

Le monoïde de Baxter

Il existe une large gamme de monoïdes définis en tant que quotients du monoïde libre et dont les classes d'équivalence sont indexées par des objets combinatoires. Le premier exemple historique et sans doute le plus fondamental, est le monoïde plaxique [LS81], [Lot02]. Une définition de ce monoïde repose sur la donnée d'une congruence sur les mots obtenue à partir des relations de Knuth [Knu70]. Une autre approche, plus algorithmique, consiste à partir de l'algorithme de Schensted [Sch61], qui permet de déterminer sur l'entrée d'un mot, la longueur de son plus long sous-mot croissant. Cet algorithme construit un effet un tableau de Young à partir d'un mot, et il est naturel de considérer que deux mots sont équivalents s'ils produisent le même tableau lorsqu'on les insère. Il s'avère que ces deux approches mènent à la construction du même monoïde, que les éléments de ce dernier peuvent s'interpréter comme des tableaux de Young et que le produit du monoïde se calcule par l'intermédiaire de l'algorithme de Schensted. La principale illustration de l'importance du monoïde plaxique est son intervention dans la démonstration de la règle de Littlewood-Richardson [LR34] pour calculer les coefficients qui apparaissent lors de la multiplication de deux fonctions de Schur [Mac95]. Il permet de plus de construire une généralisation de l'algèbre des fonctions symétriques [Mac95] en la plongeant dans l'algèbre des tableaux de Young, rendant plus limpides certaines propriétés des fonctions symétriques.

Plus récemment, d'autres monoïdes qui ont des caractéristiques semblables au monoïde plaxique ont été introduits et étudiés. L'exemple le plus convaincant est dans doute le monoïde sylvestre [HNT02], [HNT05], qui, tout comme le monoïde plaxique, peut être défini de deux manières équivalentes, soit par une congruence sur les mots, soit par un algorithme d'insertion. Dans ce cas, c'est l'algorithme d'insertion dans un arbre binaire de recherche [AU94], [Knu98], [CLRS03] qui joue le rôle de l'algorithme de Schensted, et les objets du monoïde sont des arbres binaires de recherche. De plus, tout comme dans le cas plaxique, le monoïde sylvestre est un tremplin pour la construction d'une algèbre qui généralise celle des fonctions symétriques [LR98], [HNT05]. Notons que sous certaines conditions, de tels monoïdes mènent à la construction de treillis. Par exemple, le monoïde sylvestre offre une construction très intéressante [HNT05] du treillis de Tamari [Tam62], [HT72].

L'objectif de ce chapitre est d'introduire et d'étudier un monoïde semblable au monoïde plaxique et au monoïde sylvestre, dont les classes d'équivalence sont en bijection avec les objets de la famille combinatoire de Baxter. La famille combinatoire de Baxter est constituée des classes combinatoires en bijection avec les permutations de Baxter [Bax64], qui sont des permutations qui évitent certains motifs. Citons par exemple les couples d'arbres binaires jumeaux [DG94], les quadrangulations [ABP04] et les orientations planes bipolaires [BBMF08], qui sont des objets de cette famille. Nous considérerons spécialement les couples d'arbres binaires jumeaux comme représentants de cette famille.

Ce chapitre est organisé comme suit. Dans le paragraphe 4.1, nous présentons les concepts de base sur les monoïdes semblables au monoïde plaxique. Nous passons ensuite en revue quelques exemples de tels monoïdes : le monoïde plaxique, le monoïde sylvestre, le monoïde de Bell, et enfin, le monoïde des k -reculs. La définition du monoïde de Baxter est donnée dans le paragraphe 4.2. Nous y dégageons ses premières propriétés, ainsi que les liens qu'il entretient avec certains des monoïdes sus-cités. Le paragraphe 4.3 est consacrée à construire un analogue de l'algorithme d'insertion et de la correspondance de Robinson-Schensted. Nous associons ainsi bijectivement à tout mot, un couple de couples d'arbres binaires jumeaux étiquetés soumis à certaines conditions. Cette correspondance fait intervenir deux algorithmes d'insertion dans un arbre binaire de recherche, ainsi que la combinatoire des arbres binaires croissants et décroissants. De plus, à partir d'un couple d'arbres binaires jumeaux étiqueté J , nous proposons divers algorithmes d'extraction pour calculer un mot particulier dans la classe encodée par J . Nous terminons par le paragraphe 4.4 en rappelant la construction qui à une congruence d'un treillis associe un treillis quotient. La congruence de Baxter est, comme nous le montrons, une congruence du permutoèdre. Étant donné que les classes d'équivalence de permutations sous la congruence de Baxter sont en bijection avec les couples d'arbres binaires jumeaux non étiquetés, la congruence de Baxter définit un treillis quotient du permutoèdre sur l'ensemble de ces éléments. Nous décrivons ses relations de couverture et montrons que ce treillis est très similaire au treillis de Tamari. Un nouvel objet de la famille combinatoire de Baxter est finalement introduit, à savoir les *diagrammes de Tamari doubles*, dans le but de pouvoir comparer facilement deux éléments dans ce treillis.

La plupart des résultats contenus dans ce chapitre ont été publiés dans [Gir11].

4.1 Bons monoïdes

Dans cette partie, avant de donner des exemples des monoïdes connus et qui sont à bien des rapports analogues au monoïde plaxique, nous rappelons la définition de ce qu'est un *bon monoïde* [Hiv04], à comprendre au sens « semblable au monoïde plaxique ». Nous terminons cette partie en présentant des opérations sur les bons monoïdes qui permettent d'en produire de nouveaux à partir d'anciens.

4.1.1 Définitions de base

Définition 4.1.1. Une relation d'équivalence \equiv définie sur A^* est une congruence de monoïde — ou simplement congruence si le contexte est clair — si pour tous mots $u, u', v, v' \in A^*$,

$$u \equiv u' \quad \text{et} \quad v \equiv v' \quad \text{impliquent} \quad u \cdot v \equiv u' \cdot v'. \quad (4.1.1)$$

Une congruence \equiv permet de définir une structure de *monoïde quotient* du monoïde libre, que nous notons A^*/\equiv . En effet, si $\tau : A^* \rightarrow A^*/\equiv$ est la projection canonique, l'ensemble A^*/\equiv est muni du produit \cdot défini, pour toutes classes d'équivalence \hat{x} et \hat{y} de A^*/\equiv , par

$$\hat{x} \cdot \hat{y} := \tau(x \cdot y), \quad (4.1.2)$$

où x et y sont des éléments quelconques de \hat{x} et de \hat{y} respectivement.

Définition 4.1.2. Une relation d'équivalence \equiv définie sur A^* est compatible avec la déstandardisation si pour tous mots $u, v \in A^*$,

$$u \equiv v \quad \text{si et seulement si} \quad \text{std}(u) \equiv \text{std}(v) \quad \text{et} \quad \text{ev}(u) = \text{ev}(v). \quad (4.1.3)$$

Lorsque \equiv est une congruence, nous dirons par extension que le monoïde A^*/\equiv est *compatible avec la déstandardisation* si \equiv l'est.

Cette définition signifie en particulier que tous les mots d'une classe d'équivalence d'une relation compatible avec la déstandardisation possèdent la même évaluation, et à plus forte raison, la même longueur. Remarquons ainsi que la classe d'équivalence d'une permutation est constituée uniquement de permutations. En plus de cela, ce type de relation d'équivalence possède l'attrait supplémentaire d'être plus simple à étudier. On peut en effet restreindre l'étude de la relation aux mots sans répétition de lettres.

Définition 4.1.3. Une relation d'équivalence \equiv définie sur A^* est compatible aux restrictions aux intervalles d'alphabet si pour tout intervalle I de A et tous mots $u, v \in A^*$,

$$u \equiv v \quad \text{implique} \quad u|_I \equiv v|_I. \quad (4.1.4)$$

Lorsque \equiv est une congruence, nous dirons par extension que le monoïde A^*/\equiv est *compatible avec la restriction aux intervalles d'alphabet* si \equiv l'est.

La définition suivante, due à Hivert [Hiv04], met en évidence une classe de congruences sur les mots, et ainsi du même coup une classe de monoïdes quotients du monoïde libre :

Définition 4.1.4. Soit \equiv une relation d'équivalence sur A^* . Si \equiv est une congruence de monoïde, compatible avec la déstandardisation et aux restrictions aux intervalles d'alphabet, alors, le monoïde quotient A^*/\equiv est un bon monoïde.

Il est remarquable que tous les bons monoïdes connus sont définis par l'intermédiaire d'un ensemble R de relations binaires sur A^* . Une manière pratique de construire une congruence \equiv à partir d'un tel ensemble R consiste à considérer la relation binaire \leftrightarrow définie pour tout $u, v \in A^*$ par

$$u \leftrightarrow v \quad \text{s'il existe } \curvearrowright \in R \text{ tel que } u \curvearrowright v \text{ ou } v \curvearrowright u. \quad (4.1.5)$$

Deux mots sont déclarés équivalents s'ils sont en relation pour la congruence engendrée par la clôture réflexive et transitive de \leftrightarrow . En d'autres termes, on pose $xuy \equiv xu'y$ si $u = u'$ ou s'il existe une suite de mots $v^{(1)}, \dots, v^{(\ell)}$ tels que

$$u \leftrightarrow v^{(1)} \leftrightarrow \dots \leftrightarrow v^{(\ell)} \leftrightarrow u'. \quad (4.1.6)$$

Les relations de R sont appelées *relations d'adjacence* et peuvent dans certains cas être avantageusement vues comme des *règles de réécriture* — on pourra à ce sujet consulter [BN98]. Dans ce formalisme, deux mots sont équivalents si l'on peut réécrire l'un en l'autre par utilisation de la règle \leftrightarrow .

Dans ce qui suit, les congruences que nous allons considérer vont être définies en donnant simplement l'ensemble R des relations d'adjacence. Nous appelons alors la congruence ainsi obtenue à partir de R la *congruence engendrée par R* .

Plusieurs questions peuvent se poser lorsque l'on étudie un bon monoïde A^*/\equiv . L'une des plus naturelles porte sur l'encodage de ses classes d'équivalence et d'algorithmes efficaces pour décider si deux mots sont équivalents. Une autre question est d'étudier le comportement de \equiv lorsque celle-ci est restreinte aux permutations, par exemple en comptant combien il y a de classes de permutations de taille n . De plus, il est également naturel d'observer si les classes de permutations forment un intervalle du permutoèdre, ou mieux, si la congruence de monoïde \equiv est aussi une congruence de ce treillis.

Voici une autre question classique, dont nous rappelons la définition :

Définition 4.1.5. Une relation d'équivalence \equiv définie sur A^* est compatible avec l'involution de Schützenberger si pour tous mots $u, v \in A^*$,

$$u \equiv v \quad \text{si et seulement si} \quad u^\# \equiv v^\#. \quad (4.1.7)$$

Encore dans ce cas, lorsque \equiv est une congruence, nous dirons par extension que le monoïde A^*/\equiv est compatible avec l'involution de Schützenberger si \equiv l'est.

4.1.2 Quelques exemples de bons monoïdes

Nous donnons dans cette partie quelques exemples et propriétés de base de quelques bons monoïdes connus.

Le monoïde plaxique

Le *monoïde plaxique* [LS81], [Lot02] est le quotient de A^* par la congruence \equiv_P engendrée par les relations d'adjacence $\xrightarrow{P1}$ et $\xrightarrow{P2}$ définies pour $a, b, c \in A$ par

$$acb \xrightarrow{P1} cab \quad \text{où} \quad a \leq b < c, \quad (4.1.8)$$

$$bac \xrightarrow{P2} bca \quad \text{où} \quad a < b \leq c. \quad (4.1.9)$$

On vérifie sans peine que le monoïde plaxique est un bon monoïde. Il est de plus compatible avec l'involution de Schützenberger puisque l'image de la relation $\xrightarrow{P1}$ par l'involution de Schützenberger est $\xrightarrow{P2}$ et réciproquement.

Les classes plaxiques de permutations sont en bijection avec les tableaux de Young standard [LS81], [Ful97]. Cette bijection se montre en effet par l'intermédiaire de la correspondance de Robinson-Schensted. Ainsi, les premiers cardinaux des classes plaxiques de permutations par taille sont

$$1, 1, 2, 4, 10, 26, 76, 232, 764, 2620, 9496, \quad (4.1.10)$$

et comptent également le nombre d'involutions sur $[n]$ (suite A000085 de [Slo]).

Le monoïde hypoplaxique

Le *monoïde hypoplaxique* est le quotient de A^* par la congruence \equiv_H engendrée par les relations d'adjacence $\xrightarrow{H1}$ et $\xrightarrow{H2}$ définies pour $u \in A^*$ et $a, b, c \in A$ par

$$acub \xrightarrow{H1} caub \quad \text{où} \quad a \leq b < c, \quad (4.1.11)$$

$$bua \xrightarrow{H2} bua \quad \text{où} \quad a < b \leq c. \quad (4.1.12)$$

Ce bon monoïde a été introduit par Krob et Thibon [KT97], [KT99] et étudié par Novelli [Nov98]. L'appellation « hypoplaxique » provient du fait qu'il forme un quotient du monoïde plaxique. En effet, pour tous mots u et v , si $u \equiv_P v$ alors $u \equiv_H v$.

Il est intéressant de noter que les classes de permutations sont en bijection avec les *mots binaires*, i.e., les mots sur l'alphabet $\{0, 1\}$. En effet, pour encoder une classe hypoplaxique de mots de taille n , il est nécessaire et suffisant d'encoder l'ensemble des reculs qui caractérisent cette classe. Toute classe hypoplaxique de permutations contient en effet toutes les permutations qui ont un ensemble de reculs donnés. Cet ensemble peut se représenter par le mot binaire b de longueur $n - 1$ où $b_i = 1$ si et seulement si i est un recul des éléments de la classe encodée. Les premiers cardinaux des classes hypoplaxiques de permutations par taille sont donc

$$1, 1, 2, 4, 8, 16, 32, 64, 128, 256, 512, \quad (4.1.13)$$

et comptent également le nombre de compositions de l'entier n (suite A000079 de [Slo]).

Le monoïde sylvestre

Le *monoïde sylvestre* est le quotient de A^* par la congruence \equiv_s engendrée par la relation d'adjacence \xleftrightarrow{s} définie pour $u \in A^*$ et $a, b, c \in A$ par

$$acub \xleftrightarrow{s} caub \quad \text{où} \quad a \leq b < c. \quad (4.1.14)$$

Ce bon monoïde a été introduit par Hivert, Novelli et Thibon [HNT02], [HNT03], [HNT05]. Il possède un certain nombre de propriétés intéressantes, comme le fait que la congruence sylvestre est aussi une congruence de treillis du permutoèdre, et que le treillis obtenu par ce quotient est le treillis de Tamari [HNT05].

L'analogue sylvestre de l'algorithme d'insertion plaxique est l'algorithme d'insertion dans un arbre binaire de recherche droit. En effet, deux mots sont équivalents pour la relation sylvestre si et seulement si chacun d'eux donne le même arbre binaire de recherche droit lorsqu'on les insère de droite à gauche suivant cet algorithme [AU94], [Knu98], [CLRS03]. Ainsi, le \mathbb{P} -*symbole sylvestre* d'un mot est l'arbre binaire de recherche droit obtenu en insérant ses lettres de droite à gauche.

Introduisons une version légèrement modifiée de ce monoïde que nous appelons *monoïde #-sylvestre*. Ce monoïde est le quotient de A^* par la congruence $\equiv_{\#s}$, obtenue à partir de la relation d'adjacence $\xleftrightarrow{\#s}$ définie pour $u \in A^*$ et $a, b, c \in A$ par

$$bua \xleftrightarrow{\#s} buca \quad \text{où} \quad a < b \leq c. \quad (4.1.15)$$

Cette dénomination provient du fait que le monoïde #-sylvestre est défini à partir du monoïde sylvestre en considérant l'image de la relation d'adjacence sylvestre par l'involution de Schützenberger $\#$. En effet, pour tous mots $u, v \in A^*$, $u \equiv_{\#s} v$ si et seulement si $u^\# \equiv_s v^\#$.

Une version légèrement modifiée de l'algorithme d'insertion dans un arbre binaire de recherche permet également de décider de l'équivalence de deux mots sous la relation #-sylvestre. Dans notre contexte, nous appelons cet algorithme l'*insertion en feuille*, et celui-ci se décline donc en deux versions, l'une en faveur de la relation sylvestre, et l'autre, en faveur de la relation #-sylvestre.

Algorithme : INSERTIONFEUILLE.

Entrée : un arbre binaire de recherche droit (resp. gauche) T étiqueté sur A , une lettre $a \in A$.

Sortie : T après l'insertion en feuille de a .

1. Si $T = \perp$, retourner l'arbre binaire à un nœud étiqueté par a .
2. Soit b l'étiquette de la racine de T .
3. Si $a \leq b$ (resp. $a < b$) :
 - (a) alors, insérer en feuille récursivement a dans le sous-arbre gauche de T .
 - (b) Sinon, insérer en feuille récursivement a dans le sous-arbre droit de T .

Fin.

Dans le but d'y faire référence plus loin, rappelons ici le théorème de [HNT05] reformulé dans notre contexte, et complété pour qu'il reste valable dans le cadre de la relation #-sylvestre.

Théorème 4.1.6. *Deux mots sont équivalents pour la relation sylvestre (resp. #-sylvestre) si et seulement si chacun d'eux donne le même arbre binaire de recherche droit (resp. gauche) lorsque l'on insère ses lettres de droite à gauche (resp. gauche à droite).*

En d'autres termes, tout arbre binaire de recherche droit (resp. gauche) encode une classe d'équivalence sylvestre (resp. #-sylvestre), et réciproquement.

Expliquons la partie du théorème 4.1.6 qui concerne la relation #-sylvestre. Il s'ensuit de (4.1.15) qu'encoder une classe #-sylvestre d'un mot u est équivalent à encoder la classe sylvestre de $u^\#$. Pour cela, il suffit simplement d'insérer u de gauche à droite en considérant que la relation d'ordre portant sur ses lettres est inversée. De cette manière, nous obtenons un arbre binaire tel que pour chacun de ses nœuds x étiquetés par une lettre b , toutes les étiquettes a des nœuds du sous-arbre gauche de x , et toutes les étiquettes c des nœuds du sous-arbre droit de x , l'inégalité $a \geq b > c$ est établie. Cet arbre n'est clairement pas un arbre binaire de recherche gauche. Cependant, un arbre binaire de recherche gauche peut être obtenu en échangeant, pour chacun de ses nœuds, ses sous-arbres gauches et droits. Il est possible de montrer par récurrence sur $|u|$ que cet arbre binaire de recherche gauche est effectivement celui que produit INSERTIONFEUILLE en insérant les lettres de u de gauche à droite. Ceci justifie la différence de traitement entre arbres binaires de recherche gauches et droits au niveau de l'instruction (3) de INSERTIONFEUILLE.

Ainsi, comme il existe une unique manière d'étiqueter par une permutation un arbre binaire de manière à en faire un arbre binaire de recherche, les classes sylvestres et #-sylvestres de permutations de taille n peuvent s'encoder par des arbres binaires non étiquetés à n nœuds. Les premiers cardinaux des classes sylvestres de permutations par taille sont donc

$$1, 1, 2, 5, 14, 42, 132, 429, 1430, 4862, 16796, \quad (4.1.16)$$

et comptent les objets catalans (suite A000108 de [Slo]).

Le monoïde de Bell

Le *monoïde de Bell* est le quotient de A^* par la congruence \equiv_{BL} engendré par la relation d'adjacence \xleftrightarrow{BL} définie pour $u \in A^*$ et $a, b, c \in A$ par

$$acub \xleftrightarrow{BL} caub \quad \text{où} \quad a \leq b < c \text{ et pour tout } d \in \text{Alph}(u), d \geq c. \quad (4.1.17)$$

Ce bon monoïde a été introduit en [Rey07]. Un analogue de l'algorithme d'insertion de Robinson-Schensted y est construit, et permet de montrer que les classes de permutations de taille n sous la congruence \equiv_{BL} sont en bijection avec les partitions de l'ensemble $[n]$. Les premiers cardinaux des classes de permutations par taille sont

$$1, 1, 2, 5, 15, 52, 203, 877, 4140, 21147, 115975, \quad (4.1.18)$$

et comptent les partitions de l'ensemble $[n]$ (suite A000110 de [Slo]).

Nous pouvons observer que la relation \equiv_{BL} est un raffinement de la relation sylvestre car $u \equiv_{BL} v$ implique $u \equiv_S v$ pour tout mots u et v . De plus, toute classe de permutations de \equiv_{BL} est un intervalle du permutoèdre [Rey07].

Le monoïde des k -reculs

Si a et c sont deux lettres, on note par $c - a$ le cardinal de l'ensemble $\{b \in A : a < b \leq c\}$. Dans [NRT09], Novelli, Reutenauer et Thibon définissent, pour tout $k \geq 0$, la congruence \equiv_{kR} engendrée par la relation d'adjacence \xleftrightarrow{kR} définie pour $a, c \in A$ par

$$ac \xleftrightarrow{kR} ca \quad \text{où} \quad c - a \geq k. \quad (4.1.19)$$

Cette congruence définit le *monoïde des k -reculs*, qui n'est cependant pas en l'état un bon monoïde. En effet, ce monoïde n'est pas compatible avec la déstandardisation puisque l'on a par exemple $13 \equiv_{2R} 31$ mais $12 = \text{std}(13)$ n'est pas équivalent à $21 = \text{std}(31)$.

En reformulant la relation d'adjacence \xleftrightarrow{kR} pour $u, v \in A^*$ et $a, c \in A$ par

$$u \mathbf{a} c v \xleftrightarrow{kR} u \mathbf{c} a v \quad \text{où} \quad |u|_{[a,c]} + |v|_{[a,c]} \geq k - 1, \quad (4.1.20)$$

on obtient un bon monoïde. En d'autres termes, (4.1.20) signifie que l'on peut réécrire un mot w en transposant deux lettres adjacentes si et seulement s'il existe dans w au moins $k - 1$ lettres qui se trouvent en valeur entre a et c . Dans le reste du texte, nous considérerons cette reformulation lorsque nous ferons référence au monoïde des k -reculs.

Remarquons que la congruence \equiv_{2R} est la congruence hypoplaxique et que pour tout $k \geq 0$, la relation \equiv_{k+1R} est un raffinement de \equiv_{kR} , et ainsi, le monoïde des k -reculs peut être vu comme une généralisation du monoïde hypoplaxique.

4.1.3 Opérations sur les bons monoïdes

Nous définissons dans cette partie deux opérations sur les bons monoïdes qui permettent de construire de nouveaux bons monoïdes à partir d'anciens.

Définition 4.1.7. Soient M et N deux bons monoïdes définis respectivement à partir des congruences \equiv_M et \equiv_N . Soit \equiv la relation d'équivalence définie pour tous $u, v \in A^*$ par

$$u \equiv v \quad \text{si et seulement si} \quad u \equiv_M v \quad \text{et} \quad u \equiv_N v. \quad (4.1.21)$$

L l'intersection de M et de N est le monoïde A^*/\equiv .

Proposition 4.1.8. L'intersection de deux bons monoïdes est un bon monoïde.

Démonstration. Soient $M := A^*/\equiv_M$ et $N := A^*/\equiv_N$ deux bons monoïdes, et $L := A^*/\equiv$ la structure obtenue par l'intersection de M et de N . La démonstration du fait que la relation \equiv définie en (4.1.21) est bien une relation d'équivalence ainsi qu'une congruence est directe. Par conséquent, la définition 4.1.7 est consistante puisque L est bien un monoïde.

Montrons maintenant que L est bien un bon monoïde. Soient I un intervalle de A et $u, v \in A^*$ tels que $u \equiv v$. Par définition, $u \equiv_M v$ et $u \equiv_N v$, et comme \equiv_M et \equiv_N sont compatibles aux restrictions aux intervalles d'alphabet, $u|_I \equiv_M v|_I$ et $u|_I \equiv_N v|_I$. Par conséquent, nous avons $u|_I \equiv v|_I$, et \equiv est compatible aux restrictions aux intervalles d'alphabet. De plus, comme \equiv_M et \equiv_N sont compatibles avec la déstandardisation, nous avons $\text{ev}(u) = \text{ev}(v)$ et $\text{std}(u) \equiv_M \text{std}(v)$ et $\text{std}(u) \equiv_N \text{std}(v)$. Ceci implique donc $\text{std}(u) \equiv \text{std}(v)$. Inversement, si u et v sont tels que $\text{ev}(u) = \text{ev}(v)$ et $\text{std}(u) \equiv \text{std}(v)$, alors par définition de \equiv , $\text{std}(u) \equiv_M \text{std}(v)$ et $\text{std}(u) \equiv_N \text{std}(v)$. La compatibilité avec la déstandardisation de \equiv_M et \equiv_N implique $u \equiv_M v$ et $u \equiv_N v$, ce qui entraîne $u \equiv v$ et montre que \equiv est compatible avec la déstandardisation. \square

Définition 4.1.9. Soit M un bon monoïde défini à partir de la congruence \equiv_M . Soit \equiv la relation d'équivalence définie pour tous $u, v \in A^*$ par

$$u \equiv v \quad \text{si et seulement si} \quad u^\# \equiv_M v^\#. \quad (4.1.22)$$

L l'image par l'involution de Schützenberger de M est le monoïde A^*/\equiv .

Avant de montrer que l'image par l'involution de Schützenberger d'un bon monoïde est un bon monoïde, nous avons besoin du lemme suivant, qui souligne une forme de compatibilité entre la standardisation et l'involution de Schützenberger sur les mots :

Lemme 4.1.10. L'involution de Schützenberger et la standardisation commutent. Autrement dit, pour tout $u \in A^*$, $\text{std}(u^\#) = \text{std}(u)^\#$.

Démonstration. Par définition du processus de standardisation, la permutation $\sigma := \text{std}(u)$ est l'unique permutation de taille $n := |u|$ qui possède le même ensemble d'inversions que u . De plus, par définition de l'involution de Schützenberger, (i, j) est une inversion de σ si et seulement si $(n+1-j, n+1-i)$ est une inversion de $\sigma^\#$. De même, on observe que (i, j) est une inversion de u si et seulement si $(n+1-j, n+1-i)$ est une inversion de $u^\#$. Le résultat suit du fait qu'une permutation est entièrement déterminée par son ensemble d'inversions. \square

Proposition 4.1.11. *L'image par l'involution de Schützenberger d'un bon monoïde est un bon monoïde.*

Démonstration. Soient $M := A^*/\equiv_{\mathfrak{M}}$ un bon monoïde et $N := A^*/\equiv$ l'image par l'involution de Schützenberger de M . La démonstration du fait que la relation \equiv définie en (4.1.22) est bien une relation d'équivalence est directe.

Montrons que \equiv est bien une congruence de monoïdes. Soient u, u', v et v' des mots tels que $u \equiv u'$ et $v \equiv v'$. Nous avons alors $u^\# \equiv_{\mathfrak{M}} u'^\#$ et $v^\# \equiv_{\mathfrak{M}} v'^\#$. Comme $\equiv_{\mathfrak{M}}$ est une congruence, on a

$$u^\# \cdot v^\# \equiv_{\mathfrak{M}} u'^\# \cdot v'^\#, \quad (4.1.23)$$

et ainsi,

$$(u^\# \cdot v^\#)^\# \equiv (u'^\# \cdot v'^\#)^\#, \quad (4.1.24)$$

ce qui implique $v \cdot u \equiv v' \cdot u'$ par définition de l'involution de Schützenberger, et montre que \equiv est une congruence. La définition 4.1.9 est donc consistante puisque N est bien un monoïde.

Montrons maintenant que N est un bon monoïde. Soit I un intervalle de A et supposons $u \equiv v$. Alors, $u^\# \equiv_{\mathfrak{M}} v^\#$, et comme $\equiv_{\mathfrak{M}}$ est compatible aux restrictions aux intervalles d'alphabet, $(u^\#)_{|I} \equiv_{\mathfrak{M}} (v^\#)_{|I}$. Maintenant, comme on a $(w^\#)_{|E} = (w_{|E^\#})^\#$ pour tout mot w et tout alphabet E , nous avons $(u_{|I^\#})^\# \equiv_{\mathfrak{M}} (v_{|I^\#})^\#$, et ainsi, $u_{|I^\#} \equiv v_{|I^\#}$. Étant donné que $I^\#$ est également un intervalle de A , \equiv est compatible aux restrictions aux intervalles d'alphabet.

Supposons maintenant $u \equiv v$. Alors, $u^\# \equiv_{\mathfrak{M}} v^\#$, et comme $\equiv_{\mathfrak{M}}$ est compatible avec la déstandardisation, nous avons $\text{ev}(u^\#) = \text{ev}(v^\#)$ et $\text{std}(u^\#) \equiv_{\mathfrak{M}} \text{std}(v^\#)$. Maintenant, d'après le lemme 4.1.10, on a $\text{std}(u)^\# \equiv_{\mathfrak{M}} \text{std}(v)^\#$, et ainsi, $\text{std}(u) \equiv \text{std}(v)$. Réciproquement, si $\text{ev}(u) = \text{ev}(v)$ et $\text{std}(u) \equiv \text{std}(v)$, nous avons $\text{std}(u)^\# \equiv_{\mathfrak{M}} \text{std}(v)^\#$ et $\text{ev}(u^\#) = \text{ev}(v^\#)$. On a, d'après le lemme 4.1.10, $\text{std}(u^\#) \equiv_{\mathfrak{M}} \text{std}(v^\#)$, et, comme M est compatible avec la déstandardisation, $u^\# \equiv_{\mathfrak{M}} v^\#$, ce qui implique finalement $u \equiv v$. \square

4.2 La congruence de Baxter

Nous définissons dans ce paragraphe le monoïde de Baxter et dégageons ses premières propriétés.

4.2.1 Permutations de Baxter et couples d'arbres binaires jumeaux

Rappelons qu'une *permutation de Baxter* [Bax64] est une permutation σ telle que pour tout sous-mot $u := u_1 u_2 u_3 u_4$ de σ où les lettres u_2 et u_3 sont adjacentes dans σ , on a $\text{std}(u) \notin \{2413, 3142\}$. En d'autres termes, σ est une permutation de Baxter si elle évite les *motifs généralisés de permutation* $2-41-3$ et $3-14-2$ (voir [BS00] pour une introduction sur les motifs généralisés de permutation). Par exemple, **42173856** n'est pas une permutation de Baxter puisqu'elle admet le sous-mot 4735 où les lettres 7 et 3 sont adjacentes et $\text{std}(4732) = 2413$. En revanche, 436975128 est une permutation de Baxter. Nous notons \mathfrak{S}^{B} la classe combinatoire des permutations de Baxter où la taille d'un élément est sa taille en tant que permutation.

Les cardinaux des ensembles \mathfrak{S}_n^B constituent la suite **A001181** de [Slo] qui débute par

$$1, 1, 2, 6, 22, 92, 422, 2074, 10754, 58202, 326240. \quad (4.2.1)$$

Définition 4.2.1. Un couple d'arbres binaires jumeaux est un couple (T_L, T_R) d'arbres binaires tels que la canopée de T_L est complémentaire de la canopée de T_R . La classe combinatoire des couples d'arbres binaires jumeaux est notée \mathcal{J} . La taille d'un couple d'arbres binaires jumeaux (T_L, T_R) est la taille de l'arbre binaire T_L (ou de manière équivalente, la taille de T_R).

La figure 4.1 montre un exemple de couple d'arbres binaires jumeaux.

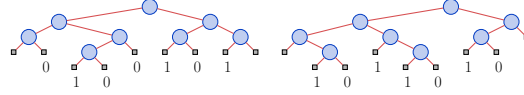


FIGURE 4.1 – Un couple d'arbres binaires jumeaux de taille 8.

Dulucq et Guibert [DG94] ont mis en évidence une bijection entre les permutations de Baxter et les couples d'arbres binaires jumeaux. La bijection que nous obtenons entre ces deux objets est très similaire.

Nous introduisons une notion de *couple d'arbres binaires de recherche jumeaux*, qui est aux couples d'arbres binaires jumeaux ce que la notion d'arbre binaire de recherche est aux arbres binaires :

Définition 4.2.2. Un couple d'arbres binaires de recherche jumeaux (T_L, T_R) est un couple d'arbres binaires jumeaux tel que T_L est un arbre binaire de recherche gauche, T_R est un arbre binaire de recherche droit, et les lectures infixes de T_L et de T_R sont les mêmes.

La figure 4.2 montre un couple d'arbres binaires de recherche jumeaux.

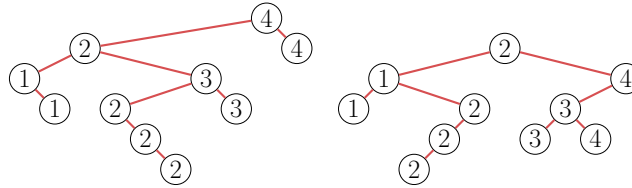


FIGURE 4.2 – Un couple d'arbres binaires de recherche jumeaux de taille 10.

La *forme* d'un couple d'arbres binaires de recherche jumeaux (T_L, T_R) est le couple d'arbres binaires jumeaux (T'_L, T'_R) tel que T'_L est la forme de T_L et T'_R est la forme de T_R .

4.2.2 La relation d'équivalence de Baxter et ses premières propriétés

Définition 4.2.3. Le monoïde de Baxter est le quotient du monoïde libre A^* par la congruence de Baxter \equiv_B obtenue à partir des relations d'adjacence de Baxter $\xrightarrow{B1}$ et $\xrightarrow{B2}$ définies pour $u, v \in A^*$ et $a, b, c, d \in A$ par

$$c u a d v b \xrightarrow{B1} c u d a v b \quad \text{où} \quad a \leq b < c \leq d, \quad (4.2.2)$$

$$b u d a v c \xrightarrow{B2} b u a d v c \quad \text{où} \quad a < b \leq c < d. \quad (4.2.3)$$

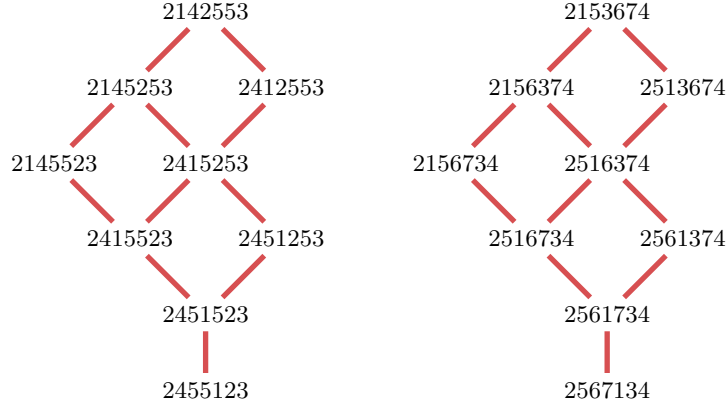


FIGURE 4.3 – La classe d'équivalence de Baxter du mot $u := 2415253$ et de la permutation $2516374 = \text{std}(u)$. Les arêtes représentent les relations d'adjacence de Baxter.

Par exemple, la classe d'équivalence de Baxter de 2415253 (voir la figure 4.3) est

$$\{2142553, 2145253, 2145523, 2412553, 2415253, 2415523, 2451253, 2451523, 2455123\}. \quad (4.2.4)$$

Notons que si l'on restreint la congruence de Baxter aux mots sans répétition de lettres, les relations d'adjacence $\xleftrightarrow{B1}$ et $\xleftrightarrow{B2}$ peuvent alors être remplacées par l'unique relation d'adjacence \xleftrightarrow{B} qui vérifie

$$x u a d v y \xleftrightarrow{B} x u d a v y \quad \text{où} \quad a < x, y < d. \quad (4.2.5)$$

Proposition 4.2.4. *Le monoïde de Baxter est compatible avec la déstandardisation.*

Démonstration. Il suffit de montrer la propriété pour les relations d'adjacence. Soient $u, v \in A^*$ tels que $u \xleftrightarrow{B1} v$. Nous avons alors $u = xcyadzbt$ et $v = xcydazbt$ avec $x, y, z, t \in A^*$ et $a \leq b < c \leq d \in A$. Comme $\xleftrightarrow{B1}$ agit en permutant des lettres, on a $\text{ev}(u) = \text{ev}(v)$. De plus, les lettres a', b', c' et d' de $\text{std}(u)$ respectivement aux mêmes positions que les lettres a, b, c et d dans u vérifient $a' < b' < c' < d'$ à cause de leurs positions relatives dans $\text{std}(u)$ et les relations d'ordre entre a, b, c et d . Les mêmes relations existent pour les lettres de $\text{std}(v)$, montrant que $\text{std}(u) \xleftrightarrow{B1} \text{std}(v)$. La démonstration pour le cas $u \xleftrightarrow{B2} v$ est analogue.

Réciproquement, supposons que v est une permutation de u et $\text{std}(u) \xleftrightarrow{B1} \text{std}(v)$. On a $\text{std}(u) = xcyadzbt$ et $\text{std}(v) = xcydazbt$ avec $x, y, z, t \in A^*$ et $a < b < c < d \in A$. Le mot u est une version non standardisée de $\text{std}(u)$ de sorte que les lettres a', b', c' et d' de u respectivement aux mêmes positions que les lettres a, b, c et d de $\text{std}(u)$ vérifient $a' \leq b' < c' \leq d'$ à cause de leurs positions relatives dans u et les relations d'ordre entre a, b, c et d . Les mêmes relations existent pour les lettres de v , montrant que $u \xleftrightarrow{B1} v$. La démonstration pour le cas $u \xleftrightarrow{B2} v$ est analogue. \square

Proposition 4.2.5. *Le monoïde de Baxter est compatible aux restrictions aux intervalles d'alphabet.*

Démonstration. Il suffit de montrer la propriété pour les relations d'adjacence. De plus, par la proposition 4.2.4, il suffit de montrer la propriété pour les permutations. Soient $\sigma, \nu \in \mathfrak{S}_n$ tels que $\sigma \xleftrightarrow{B} \nu$. Nous avons $\sigma = txuadvyw$ et $\nu = txudavyw$ avec $a < x, y < d$. Soit I un intervalle de $\{1, \dots, n\}$ et $R := I \cap \{a, x, y, d\}$. Si $R = \{a, x, y, d\}$, alors

$$\sigma|_I = t|_I x u|_I a d v|_I y w|_I \quad \text{et} \quad \nu|_I = t|_I x u|_I d a v|_I y w|_I, \quad (4.2.6)$$

de sorte que $\sigma|_I \xleftrightarrow{B} \nu|_I$. Autrement, on a $\sigma|_I = \nu|_I$ et ainsi, $\sigma|_I \equiv_B \nu|_I$. \square

Proposition 4.2.6. *Le monoïde de Baxter est compatible avec l'involution de Schützenberger.*

Démonstration. Il suffit de montrer la propriété pour les relations d'adjacence. De plus, par la proposition 4.2.4, il suffit de montrer la propriété pour les permutations. Soient $\sigma, \nu \in \mathfrak{S}_n$ tels que $\sigma \xleftrightarrow{B} \nu$. Nous avons $\sigma = txuadvyw$ et $\nu = txudavyw$ avec $a < x, y < d$. Nous avons

$$\sigma^\# = w^\# y^\# v^\# d^\# a^\# u^\# x^\# t^\# \quad \text{et} \quad \nu^\# = w^\# y^\# v^\# a^\# d^\# u^\# x^\# t^\#. \quad (4.2.7)$$

Étant donné que $d^\# < x^\#, y^\# < a^\#$, nous avons $\sigma^\# \xleftrightarrow{B} \nu^\#$. \square

4.2.3 Liens avec d'autres bons monoïdes connus

Nous mettons ici en évidence un lien entre le monoïde de Baxter et les monoïdes sylvestre et $\#$ -sylvestre. Ce lien, riche en conséquences pour le reste du chapitre, permet de montrer que le monoïde de Baxter s'obtient à partir du monoïde sylvestre en l'intersectant avec son image par l'involution de Schützenberger. L'une des premières applications de cette propriété apparaît dans la démonstration du fait que les classes d'équivalence de Baxter de permutations forment des intervalles du permutoèdre. Nous donnons aussi un lien entre le monoïde de Baxter et le monoïde des 3-reculs.

Lien avec les monoïdes sylvestre et $\#$ -sylvestre

Montrons pour commencer une propriété sur les classes sylvestres de permutations qui nous sera utile par la suite.

Lemme 4.2.7. *Soient $u := xac y$ et $v := xca y$ deux mots tels que $x, y \in A^*$, $a < c \in A$ et $u \equiv_s v$. Alors, $u \xleftrightarrow{S} v$.*

Démonstration. Ce lemme est une conséquence du théorème 4.1.6. Comme u et v donnent le même arbre binaire de recherche droit T en les insérant de droite à gauche, le nœud étiqueté par a et le nœud étiqueté par c dans T ne peuvent pas être ancêtres l'un de l'autre. Ceci implique qu'il existe un nœud étiqueté par une lettre b , ancêtre commun aux nœuds étiquetés par a et par c tel que $a \leq b < c$. Ainsi, $u \xleftrightarrow{S} v$. \square

Le lemme 4.2.7 montre également que les relations d'adjacence \xleftrightarrow{S} de toute classe sylvestre C de permutations sont exactement les relations de couverture du permutoèdre restreint aux éléments de C . Notons qu'une démonstration analogue du lemme 4.2.7 montre que c'est aussi le cas pour la relation d'adjacence $\xleftrightarrow{\#S}$ du monoïde $\#$ -sylvestre.

Le monoïde de Baxter, le monoïde sylvestre, et le monoïde $\#$ -sylvestre sont liés de la manière suivante :

Proposition 4.2.8. *Soient $u, v \in A^*$. Alors, $u \equiv_B v$ si et seulement si $u \equiv_s v$ et $u \equiv_{\#s} v$.*

Démonstration. Une fois de plus, il suffit de montrer la propriété pour les relations d'adjacence de Baxter. De plus, par la proposition 4.2.4, il suffit de montrer la propriété pour les permutations. Soient $\sigma, \nu \in \mathfrak{S}_n$ tels que $\sigma \xleftrightarrow{B} \nu$. Nous avons $\sigma = xxyadz y t$ et $\nu = xxydzay t$ avec $a < x, y < d$. La présence des lettres a, d et y avec $a < y < d$ assure $\sigma \xleftrightarrow{S} \nu$. En outre, la présence des lettres x, a et d avec $a < x < d$ assure $\sigma \xleftrightarrow{\#S} \nu$.

Réciproquement, étant donné que les monoïdes sylvestre et $\#$ -sylvestre sont compatibles avec la déstandardisation [HNT05], il suffit de montrer la propriété pour les permutations.

Soient $\sigma, \nu \in \mathfrak{S}_n$ tels que $\sigma \equiv_s \nu$ et $\sigma \equiv_{\#s} \nu$. Posons $\tau := \inf_{\leq_P} \{\sigma, \nu\}$. Comme le permutoèdre est un treillis, τ est bien défini, et comme les classes d'équivalence sylvestres et $\#$ -sylvestres de permutations sont des intervalles du permutoèdre [HNT05], nous avons $\sigma \equiv_s \tau \equiv_s \nu$ et $\sigma \equiv_{\#s} \tau \equiv_{\#s} \nu$. De plus, par le lemme 4.2.7, et encore par le fait que les classes d'équivalence sylvestres et $\#$ -sylvestres de permutations sont des intervalles du permutoèdre, pour toute chaîne saturée $\tau \leq_P \pi \leq_P \dots \leq_P \sigma$ et $\tau \leq_P \mu \leq_P \dots \leq_P \nu$, il existe les suites de relations d'adjacence $\tau \xrightarrow{s} \pi \xrightarrow{s} \dots \xrightarrow{s} \sigma$, $\tau \xrightarrow{\#s} \pi \xrightarrow{\#s} \dots \xrightarrow{\#s} \sigma$, $\tau \xrightarrow{s} \mu \xrightarrow{s} \dots \xrightarrow{s} \nu$ et $\tau \xrightarrow{\#s} \mu \xrightarrow{\#s} \dots \xrightarrow{\#s} \nu$. Par conséquent, $\tau \equiv_B \sigma$ et $\tau \equiv_B \nu$, ce qui implique $\sigma \equiv_B \nu$. \square

La proposition 4.2.8 montre que les classes d'équivalence de Baxter sont des intersections de classes d'équivalences sylvestre et $\#$ -sylvestres.

À partir de la caractérisation des classes d'équivalence de Baxter fournie par la proposition 4.2.8 et en restreignant la relation \equiv_B aux permutations, on obtient :

Proposition 4.2.9. *Pour tout $n \geq 0$, tout classe d'équivalence de $\mathfrak{S}_n / \equiv_B$ est un intervalle du permutoèdre.*

Démonstration. Par la proposition 4.2.8, les classes de Baxter sont des intersections de classes sylvestres et $\#$ -sylvestres. De plus, les classes d'équivalence de permutations sous les relations sylvestres et $\#$ -sylvestres sont des intervalles du permutoèdre [HNT05]. La proposition provient du fait que l'intersection de deux intervalles dans un treillis est également un intervalle. \square

Lemme 4.2.10. *Soient $u := x \mathbf{a} \mathbf{d} \mathbf{y}$ et $v := x \mathbf{d} \mathbf{a} \mathbf{y}$ tels que $x, y \in A^*$, $\mathbf{a} < \mathbf{d} \in A$ et $u \equiv_B v$. Alors, $u \xrightarrow{B1} v$ ou $u \xrightarrow{B2} v$.*

Démonstration. D'après la proposition 4.2.8, étant donné que $u \equiv_B v$, nous avons $u \equiv_s v$ et ainsi, par le lemme 4.2.7, $u \xrightarrow{s} v$. Ceci implique l'existence d'une lettre \mathbf{y} dans le facteur \mathbf{y} vérifiant $\mathbf{a} \leq \mathbf{y} < \mathbf{d}$. De la même manière, nous avons aussi $u \equiv_{\#s} v$ et ainsi $u \xrightarrow{\#s} v$. Ceci implique l'existence d'une lettre \mathbf{x} dans le facteur \mathbf{x} vérifiant $\mathbf{a} < \mathbf{x} \leq \mathbf{d}$. Par conséquent, u et v sont adjacents par $\xrightarrow{B1}$ ou $\xrightarrow{B2}$. \square

Le lemme 4.2.10 est l'analogue, dans le cas de la relation d'équivalence de Baxter, du lemme 4.2.7 et montre aussi que les relations d'adjacence de Baxter de toute classe de Baxter C de permutations sont exactement les relations de couverture du permutoèdre restreint aux éléments de C .

Lien avec le monoïde des 3-reculs

Le monoïde de Baxter et le monoïde des 3-reculs sont liés de la manière suivante :

Proposition 4.2.11. *Toute classe d'équivalence de \equiv_{3R} peut être exprimée comme une union de classes de Baxter.*

Démonstration. Ceci revient à montrer que pour tous mots u et v , $u \equiv_B v$ implique $u \equiv_{3R} v$. D'après la proposition 4.2.4, il suffit de montrer cette propriété sur les permutations. Supposons ainsi $\sigma \equiv_B \nu$ où σ et ν sont des permutations. Il suffit maintenant de montrer la propriété pour les relations d'adjacence de Baxter. Supposons par conséquent que $\sigma \xrightarrow{B1} \nu$. Nous avons $\sigma = txuadvyw$ et $\nu = txudavyw$ avec $\mathbf{a} < \mathbf{x}, \mathbf{y} < \mathbf{d}$. Comme σ et ν sont des permutations, $\mathbf{x} \neq \mathbf{t}$ et ainsi, $\mathbf{a} < \mathbf{x} < \mathbf{y} < \mathbf{d}$ ou $\mathbf{a} < \mathbf{y} < \mathbf{x} < \mathbf{d}$. Il existe donc au minimum les deux lettres \mathbf{x} et \mathbf{y} qui se situent en valeur entre \mathbf{a} et \mathbf{d} dans σ et dans ν . On en conclut ainsi que $\sigma \equiv_{3R} \nu$. \square

Observons que la proposition 4.2.11 n'est plus vraie pour la relation \equiv_{4R} puisqu'il y a 22 classes d'équivalence de permutations de taille 4 pour \equiv_B mais 24 pour \equiv_{4R} . Réciproquement, la relation \equiv_{4R} n'est pas non plus un raffinement de \equiv_B puisque pour tout $n \geq 5$, la permutation $1 \cdot n \cdot n-1 \dots 2$ est la seule de sa classe de Baxter mais pas de sa classe pour \equiv_{4R} .

De plus, étant donné que les classes d'équivalence de \equiv_{kR} sont des unions de classes de \equiv_{k+1R} , d'après la proposition 4.2.11, les classes hypoplaxiques sont des unions de classes de Baxter. Le monoïde de Baxter peut donc être vu comme une généralisation du monoïde hypoplaxique.

4.3 Une correspondance à la Robinson-Schensted

Nous décrivons un algorithme permettant de décider de l'équivalence de deux mots conformément à la congruence de Baxter. Étant donné un mot $u \in A^*$, cet algorithme produit son \mathbb{P} -symbole, un couple d'arbres binaires de recherche jumeaux (T_L, T_R) (voir la définition 4.2.2) tel que les lectures infixes de T_L et de T_R donnent toutes deux le réarrangement croissant de u . Il produit aussi son \mathbb{Q} -symbole, un couple d'arbres binaires jumeaux (S_L, S_R) tel que S_L (resp. S_R) est un arbre binaire croissant (resp. décroissant), et les lectures infixes de S_L et de S_R sont les mêmes.

La principale motivation d'introduire cet algorithme, en plus du fait qu'il fournit un analogue de l'algorithme classique d'insertion dans un tableau de Young, est qu'il permet de montrer que les classes de Baxter de permutations de taille n sont en bijection avec les couples d'arbres binaires jumeaux non étiquetés, et donc, par les résultats de Dulucq et Guibert [DG94], également en bijection avec les permutations de Baxter de taille n . L'autre motivation est de fournir un analogue de la correspondance de Robinson-Schensted en montrant que l'application $u \mapsto (\mathbb{P}(u), \mathbb{Q}(u))$ est une bijection entre les mots de A^* et l'ensemble des couples $((T_L, T_R), (S_L, S_R))$ décrits ci-dessus.

Nous allons commencer par établir le fait que pour tout mot u , le \mathbb{P} -symbole de u respecte les conditions que nous venons d'énoncer. Ensuite, nous montrons que l'on a bien $u \equiv_B v$ si et seulement si $\mathbb{P}(u) = \mathbb{P}(v)$, ce qui établira une injection entre l'ensemble des classes de Baxter de permutations de taille n et l'ensemble des couples d'arbres binaires jumeaux de taille n . La surjection entre ces deux ensembles va suivre d'un algorithme d'extraction qui, étant donné un couple d'arbres binaires jumeaux, calcule l'unique permutation de Baxter de sa classe. Nous donnons de plus un algorithme qui extrait la plus petite permutation de sa classe, ce qui mène à un nouvel algorithme pour calculer la bijection entre les permutations de Baxter et les *permutations de Baxter tordues* de Reading [Rea04], [LR10].

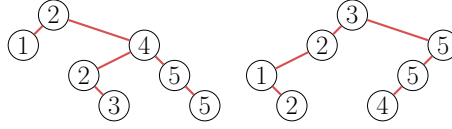
Nous terminons cette partie en donnant une version itérative de l'algorithme du \mathbb{P} -symbole, qui a pour mérite de permettre de calculer dans le monoïde de Baxter.

4.3.1 Un algorithme d'insertion à la Robinson-Schensted

Définition 4.3.1. *Le \mathbb{P} -symbole de Baxter — ou simplement \mathbb{P} -symbole lorsque le contexte est clair — d'un mot $u \in A^*$ est le couple (T_L, T_R) où T_L (resp. T_R) est l'arbre binaire de recherche gauche (resp. droit) obtenu en insérant en feuille les lettres de u de gauche à droite (resp. droite à gauche).*

À titre d'exemple, la figure 4.4 donne le \mathbb{P} -symbole du mot 2415253.

Avant de donner la démonstration du fait que ce \mathbb{P} -symbole respecte les conditions que nous avons énoncées dans l'introduction de cette partie, donnons une explication intuitive de sa validité.

FIGURE 4.4 – Le \mathbb{P} -symbole de Baxter du mot 2415253.

Soit u un mot. D'après le théorème 4.1.6, l'arbre binaire de recherche gauche T_L obtenu par des insertions en feuille des lettres de u de la gauche vers la droite encode sa classe $\#$ -sylvestre, et que, de la même manière, l'arbre binaire de recherche droit T_R obtenu par des insertions en feuille des lettres de u de la droite vers la gauche encode sa classe sylvestre. Tel que nous l'avons défini, le \mathbb{P} -symbole de u encode à la fois sa classe sylvestre et sa classe $\#$ -sylvestre. De plus, en se basant sur la proposition 4.2.8, on observe que deux mots u et v vérifient $u \equiv_{\mathbb{B}} v$ si et seulement si $u \equiv_{\mathbb{S}} v$ et $u \equiv_{\# \mathbb{S}} v$. Ainsi, l'information contenue dans le \mathbb{P} -symbole d'un mot encode exactement sa classe de Baxter.

Soit T un arbre binaire. Nous appelons i^{e} nœud de T le i^{e} nœud parcouru lors du parcours infixe de T . De même, nous appelons i^{e} feuille de T la feuille en position i lorsque l'on considère les feuilles de T de la gauche vers la droite.

La validité du \mathbb{P} -symbole de Baxter repose sur les deux lemmes et la proposition suivante.

Lemme 4.3.2. *Soit T un arbre binaire non vide et x sa i^{e} feuille. Si x est orientée vers la gauche, alors x est attachée au i^{e} nœud interne de T . Si x est orientée vers la droite, alors x est attachée au $(i-1)^{\text{e}}$ nœud de T .*

Démonstration. Nous procédons par induction structurelle sur l'ensemble des arbres binaires non vides. Si T est l'arbre binaire à un nœud, le lemme est vérifié. Autrement, on a $T = A \wedge B$. Soit x la i^{e} feuille de T et y le nœud auquel x est attachée. Si x se trouve aussi dans A et $A = \perp$, alors x est orientée vers la gauche et est attachée à la racine de T , et le lemme est vérifié car la racine de T est dans ce cas le premier nœud de T . Si x est dans A et $A \neq \perp$, x est également la i^{e} feuille de A et y est dans A , de sorte que le lemme s'ensuit par hypothèse d'induction sur A . Sinon, x est dans B . Si $B = \perp$, x est orientée vers la droite et est attachée à la racine de T , et le lemme est vérifié car la racine de T est dans ce cas le dernier nœud de T . Autrement, x est la $(i-n-1)^{\text{e}}$ feuille de B où n est le nombre de nœuds de A . Supposons que y est le j^{e} nœud de T . Alors, y est le $(j-n-1)^{\text{e}}$ nœud de B . Par conséquent, le lemme s'ensuit par hypothèse d'induction sur B . \square

Lemme 4.3.3. *Soit σ une permutation et T l'arbre binaire de recherche obtenu en insérant les lettres de σ de la gauche vers la droite. Alors, la $(i+1)^{\text{e}}$ feuille de T est orientée vers la droite si et seulement si i est un recul de σ .*

Démonstration. Posons $\mathbf{a} := i$ et $\mathbf{b} := i+1$. Supposons que \mathbf{a} est un recul de σ . Dans ce cas, σ est de la forme $\sigma = ubvaw$. Étant donné qu'aucune lettre \mathbf{x} des mots u et v ne vérifie $\mathbf{a} < \mathbf{x} < \mathbf{b}$, le nœud de T étiqueté par \mathbf{b} possède un nœud étiqueté par \mathbf{a} dans son sous-arbre gauche, lui-même n'ayant pas de sous-arbre droit et ainsi contribue, par le lemme 4.3.2, à une feuille orientée vers la droite en position $i+1$.

Réciproquement, supposons que \mathbf{a} n'est pas un recul de σ . Dans ce cas, σ est de la forme $\sigma = uavbw$. Pour la même raison que dans le cas précédent, le nœud de T étiqueté par \mathbf{a} possède un nœud étiqueté par \mathbf{b} dans son sous-arbre droit, lui-même n'ayant pas de sous-arbre gauche, et ainsi contribue, par le lemme 4.3.2, à une feuille orientée vers la gauche en position $i+1$. \square

Proposition 4.3.4. *Pour tout mot $u \in A^*$, le \mathbb{P} -symbole (T_L, T_R) de u est un couple d'arbre binaires de recherche jumeaux tel que les lectures infixes de T_L et T_R donnent toutes deux le réarrangement croissant de u .*

Démonstration. Notons que par définition de l'algorithme INSERTIONFEUILLE, T_L (resp. T_R) est un arbre binaire de recherche gauche (resp. droit), et que les lectures infixes de T_L et T_R donnent toutes deux le réarrangement croissant de u . Étant donné que les insertions en feuille des lettres de u et de $\text{std}(u)$ de la gauche vers la droite (resp. de la droite vers la gauche) dans un arbre binaire de recherche gauche (resp. droit) donnent des arbres binaires de même forme, nous pouvons considérer que $u =: \sigma$ est une permutation. Ainsi, le lemme 4.3.3 implique que les canopées de T_L et de T_R sont complémentaires puisque i est un recul de σ si et seulement si i n'est pas un recul de σ^\sim . Ceci montre que (T_L, T_R) est un couple d'arbres binaires jumeaux. \square

Théorème 4.3.5. *Soient $u, v \in A^*$. Alors, $u \equiv_{\mathbb{B}} v$ si et seulement si $\mathbb{P}(u) = \mathbb{P}(v)$.*

Démonstration. Supposons $u \equiv_{\mathbb{B}} v$. Par la proposition 4.2.8, u et v sont équivalents pour les congruences sylvestre et $\#$ -sylvestre. Par conséquent, d'après le théorème 4.1.6, u et v ont le même \mathbb{P} -symbole sylvestre et $\#$ -sylvestre, et ainsi, $\mathbb{P}(u) = \mathbb{P}(v)$.

Réciproquement, supposons $\mathbb{P}(u) = \mathbb{P}(v) =: (T_L, T_R)$. Comme les insertions en feuille de u et de v de la gauche vers la droite donnent toutes deux T_L , nous avons, par le théorème 4.1.6, $u \equiv_{\#s} v$. En outre, les insertions en feuille de u et de v de la droite vers la gauche donnent toutes deux T_R , de sorte que, par le théorème 4.1.6, $u \equiv_s v$. Ainsi, d'après la proposition 4.2.8, $u \equiv_{\mathbb{B}} v$. \square

Remarquons qu'une classe d'équivalence de Baxter de permutations peut être encodée par un couple non étiqueté d'arbres binaires jumeaux. Il existe en effet un unique moyen d'étiqueter bijectivement un arbre binaire à n nœuds sur l'ensemble $[n]$ de sorte qu'il soit un arbre binaire de recherche, et à plus forte raison, il existe un unique moyen d'étiqueter ainsi chacun des arbres d'un couple d'arbres binaires jumeaux. Par conséquent, dans ce qui suit, les couples non étiquetés d'arbres binaires jumeaux seront vus comme des couples d'arbres binaires jumeaux étiquetés par une permutation.

4.3.2 Une correspondance à la Robinson-Schensted

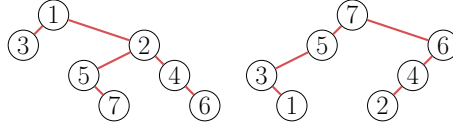
Avant de définir un \mathbb{Q} -symbole nécessaire pour établir un analogue de la correspondance de Robinson-Schensted, rappelons deux constructions classiques qui associent à un mot u un arbre binaire. L'arbre binaire croissant $\text{croi}(u)$ de u est défini récursivement par

$$\text{croi}(u) := \begin{cases} \perp & \text{si } u = \epsilon, \\ \text{croi}(v) \wedge_{\mathbf{a}} \text{croi}(w) & \text{où } u = v\mathbf{a}w, \mathbf{a} = \min \text{Alph}(u), \\ & \text{et } \mathbf{a} < \mathbf{b} \text{ pour tout } \mathbf{b} \in \text{Alph}(v). \end{cases} \quad (4.3.1)$$

Quant à lui, l'arbre binaire décroissant $\text{decr}(u)$ de u est défini récursivement par

$$\text{decr}(u) := \begin{cases} \perp & \text{si } u = \epsilon, \\ \text{decr}(v) \wedge_{\mathbf{b}} \text{decr}(w) & \text{où } u = v\mathbf{b}w, \mathbf{b} = \max \text{Alph}(u), \\ & \text{et } \mathbf{a} < \mathbf{b} \text{ pour tout } \mathbf{b} \in \text{Alph}(w). \end{cases} \quad (4.3.2)$$

Définition 4.3.6. *Le \mathbb{Q} -symbole de Baxter — ou simplement \mathbb{Q} -symbole lorsque le contexte est clair — d'un mot $u \in A^*$ est le couple (S_L, S_R) où $S_L := \text{croi}(\text{std}(u)^{-1})$ et $S_R := \text{decr}(\text{std}(u)^{-1})$.*

FIGURE 4.5 – Le \mathbb{Q} -symbole de Baxter du mot 2415253.

À titre d'exemple, la figure 4.5 représente le \mathbb{Q} -symbole du mot 2415253, dont l'inverse du standardisé est 3157246.

Il est immédiat, étant donné un mot u , que le \mathbb{Q} -symbole de u permet, en association avec son \mathbb{P} -symbole, de reconstruire u . En effet, si $\mathbb{P}(u) = (T_L, T_R)$ et $\mathbb{Q}(u) = (S_L, S_R)$, le couple (T_R, S_R) constitue à lui seul les \mathbb{P} et \mathbb{Q} -symboles sylvestres de u [HNT05], et ce couple suffit pour reconstruire u . Pour calculer le mot correspondant au couple $((T_L, T_R), (S_L, S_R))$, il suffit de lire les étiquettes de T_R dans l'ordre imposé par les étiquettes des nœuds correspondants de S_R . La même propriété est vraie pour T_L et S_L .

Pour montrer l'analogie de la correspondance de Robinson-Schensted pour le monoïde de Baxter, nous avons besoin du lemme et de la proposition suivante. Le lemme est tiré de [HNT05], reformulé dans notre contexte, et complété dans le cas des arbres binaires croissants.

Lemme 4.3.7. *Soit u un mot et $\sigma := \text{std}(u)^{-1}$. L'arbre binaire de recherche droit (resp. gauche) obtenu en insérant les lettres de u de la droite vers la gauche (resp. de la gauche vers la droite) et $\text{decr}(\sigma)$ (resp. $\text{croi}(\sigma)$) ont la même forme.*

Proposition 4.3.8. *Pour tout mot $u \in A^*$, la forme du \mathbb{Q} -symbole (S_L, S_R) de u est un couple d'arbres binaires jumeaux. De plus, S_L est un arbre binaire croissant, S_R un arbre binaire décroissant, et leur lecture infixe est la même et égale à $\text{std}(u)^{-1}$.*

Démonstration. Par définition du \mathbb{Q} -symbole, S_L et S_R sont respectivement les arbres binaires croissants et décroissants de $\sigma := \text{std}(u)^{-1}$. D'après le lemme 4.3.7, un arbre binaire de la même forme que S_L (resp. S_R) peut être obtenu en insérant en feuille les lettres de σ^{-1} de gauche à droite (resp. de droite à gauche). Ainsi, par le lemme 4.3.3, la forme de (S_L, S_R) est un couple d'arbres binaires jumeaux. De plus, par définition des constructions croi et decr , il suit par récurrence sur la taille de σ que S_L et S_R admettent bien σ comme lecture infixe. \square

Théorème 4.3.9. *L'application $u \mapsto (\mathbb{P}(u), \mathbb{Q}(u))$ définit une bijection entre les éléments de A^* et l'ensemble des couples $((T_L, T_R), (S_L, S_R))$ où*

- (i) (T_L, T_R) est un couple d'arbres binaires de recherche jumeaux;
- (ii) (S_L, S_R) est un couple d'arbres binaires jumeaux tel que S_L (resp. S_R) est un arbre binaire croissant (resp. décroissant), et S_L et S_R possèdent la même lecture infixe;
- (iii) (T_L, T_R) et (S_L, S_R) sont de même forme.

Démonstration. Commençons par montrer que pour tout $u \in A^*$, le couple $(\mathbb{P}(u), \mathbb{Q}(u))$ vérifie les trois points du théorème. Il suit de la proposition 4.3.4 que (i) est vérifié, et de la proposition 4.3.8 que (ii) l'est aussi. De plus, par le lemme 4.3.7, (iii) est établi. En outre, comme nous l'avons mentionné et justifié précédemment, il est possible de reconstruire de manière unique u à partir du couple $(\mathbb{P}(u), \mathbb{Q}(u))$. Ceci montre que la correspondance est bien définie et qu'elle est injective.

Réciproquement, supposons que $((T_L, T_R), (S_L, S_R))$ vérifie les trois points du théorème. D'après [HNT02], il existe une bijection entre les éléments de A^* et les couples (T_R, S_R) où T_R est un arbre binaire de recherche droit et S_R un arbre binaire décroissant de la même forme.

Soit u le mot en bijection avec (T_R, S_R) . De la même manière, il existe une bijection entre les éléments de A^* et les couples (T_L, S_L) où T_L est un arbre binaire de recherche gauche et S_L un arbre binaire croissant de la même forme. Soit v le mot en bijection avec (T_L, S_L) . Par hypothèse, T_L et T_R ont tous les deux la même lecture infixe, ce qui implique $\text{ev}(u) = \text{ev}(v)$. De la même manière, comme S_L et S_R ont tous les deux la même lecture infixe, $\text{std}(u)^{-1} = \text{std}(v)^{-1}$ et par conséquent, $u = v$. Observons de plus que le couple (T_L, S_L) est entièrement déterminé par le couple (T_R, S_R) et inversement. Maintenant, toujours d'après [HNT02], le couple (T_R, S_R) est le \mathbb{P} -symbole sylvestre de u , et le couple (T_L, S_R) est \mathbb{P} -symbole $\#$ -sylvestre de u . Ainsi, l'insertion de u donne le couple $((T_L, T_R), (S_L, S_R))$, montrant que l'application du théorème est également surjective. \square

4.3.3 Extraction de mots

Avant de présenter trois algorithmes qui permettent d'extraire des permutations et des mots particuliers à partir d'un couple d'arbres binaires de recherche jumeaux, commençons par une remarque sur les classes d'équivalence de Baxter qui nous sera utile par la suite.

Les classes de Baxter en tant qu'extensions linéaires

Soit T un arbre binaire étiqueté à n nœuds. Notons par $\Delta(T)$ (resp. $\nabla(T)$) le poset (N, \leq) où N est l'ensemble des nœuds de T , et \leq est défini pour $x, y \in N$ par

$$x \leq y \quad \text{si} \quad x \text{ est un ancêtre (resp. un descendant) de } y. \quad (4.3.3)$$

Si la séquence x_1, \dots, x_n est une extension linéaire de $\Delta(T)$ (resp. $\nabla(T)$), nous dirons également par abus de langage que le mot $u_1 \dots u_n$ est une *extension linéaire* de $\Delta(T)$ (resp. $\nabla(T)$) si pour tout $i \in [n]$, l'étiquette du nœud x_i est u_i .

La Note 4 de [HNT05] nous renseigne sur le fait que les mots d'une classe sylvestre encodée par un arbre binaire de recherche droit T_R sont exactement les extensions linéaires de $\nabla(T_R)$. Cette propriété est adaptable dans le contexte des classes $\#$ -sylvestres, et implique que les mots d'une classe $\#$ -sylvestre encodée par un arbre binaire de recherche gauche T_L sont exactement les extensions linéaires de $\Delta(T_L)$. De plus, d'après la proposition 4.2.8, les mots d'une classe de Baxter figurent à la fois dans une même classe sylvestre et une même classe $\#$ -sylvestre. Ainsi, les mots d'une classe de Baxter encodée par un couple d'arbres binaires de recherche jumeaux (T_L, T_R) sont exactement les mots qui sont à la fois des extensions linéaires de $\Delta(T_L)$ et de $\nabla(T_R)$.

Nous notons $\Delta(T_L) \cup \nabla(T_R)$ le poset qui regroupe les relations d'ordre de $\Delta(T_L)$ et de $\nabla(T_R)$ de la manière suivante. On considère comme égaux deux nœuds x de T_L et x' de T_R s'ils se trouvent en même position dans leurs arbres binaires respectifs. Le poset $\Delta(T_L) \cup \nabla(T_R)$ est défini sur l'ensemble des nœuds de T_L (ou de manière équivalente avec l'identification précédente, sur les nœuds de T_R) par la relation d'ordre \leq définie par

$$x \leq y \quad \text{si} \quad x \leq y \text{ dans } \Delta(T_L) \quad \text{ou} \quad x \leq y \text{ dans } \nabla(T_R). \quad (4.3.4)$$

Remarquons qu'il s'agit bien là d'un poset, *i.e.*, la relation d'ordre définie en (4.3.4) est bien antisymétrique. En effet, si l'on suppose par l'absurde que l'on a $x \leq y$ dans $\Delta(T_L)$ et $y \leq x$ dans $\nabla(T_R)$, cela signifie que x est un ancêtre de y à la fois dans T_L et dans T_R . Ceci ne peut jamais arriver car, d'après la définition 4.3.1, le \mathbb{P} -symbole (T_L, T_R) est défini de telle sorte que T_L soit obtenu par une insertion en feuille de gauche à droite d'un mot, et T_R , de droite à gauche de ce même mot.

On observe que les mots qui sont à la fois des extensions linéaires de $\Delta(T_L)$ et de $\nabla(T_R)$ sont également des extensions linéaires de $\Delta(T_L) \cup \nabla(T_R)$, et réciproquement. Ainsi, les mots

qui figurent dans la classe de Baxter encodée par (T_L, T_R) sont les extensions linéaires du poset $\Delta(T_L) \cup \nabla(T_R)$.

Par exemple, pour le couple d'arbres binaires de recherche jumeaux suivant :

$$(T_L, T_R) := \left(\begin{array}{c} \text{Arbre } T_L: \text{ racine 5, fils 2 (gauche) et 7 (droite). 2 a pour fils 1 (gauche) et 4 (droite). 4 a pour fils 3 (gauche).} \\ \text{Arbre } T_R: \text{ racine 6, fils 3 (gauche) et 7 (droite). 3 a pour fils 1 (gauche) et 4 (droite). 4 a pour fils 2 (gauche) et 5 (droite).} \end{array} \right), \quad (4.3.5)$$

l'ensemble des mots qui sont dans la classe de Baxter encodée par (T_L, T_R) est

$$\{5214376, 5214736, 5217436, 5241376, 5241736, 5247136, 5271436, 5274136, 5721436, 5724136\}, \quad (4.3.6)$$

qui est exactement l'ensemble des mots qui sont à la fois des extensions linéaires de $\Delta(T_L)$ et de $\nabla(T_R)$.

Pour construire le diagramme de Hasse du poset $\Delta(T_L) \cup \nabla(T_R)$, il suffit de partir de T_L et de lui ajouter les arcs encodant les relations d'ordre induites par T_R qui n'existent pas déjà dans celles induites par T_L .

Ainsi, pour l'exemple précédent, nous obtenons

$$\Delta(T_L) \cup \nabla(T_R) = \begin{array}{c} \text{Diagramme de Hasse:} \\ \text{Nœuds: 1, 2, 3, 4, 5, 6, 7.} \\ \text{Arcs: } 1 \rightarrow 2, 1 \rightarrow 3, 2 \rightarrow 4, 3 \rightarrow 4, 4 \rightarrow 5, 4 \rightarrow 6, 5 \rightarrow 7, 6 \rightarrow 7. \end{array}, \quad (4.3.7)$$

et l'ensemble (4.3.6) contient exactement les extensions linéaires de ce poset.

Extraction de permutations de Baxter

L'algorithme suivant permet de calculer un mot qui appartient à la classe de Baxter encodée par un couple d'arbres binaires de recherche jumeaux (T_L, T_R) en entrée. Lorsque (T_L, T_R) est étiqueté par une permutation, notre algorithme coïncide avec l'algorithme que Dulucq et Guibert ont utilisé pour présenter leur bijection entre les couples d'arbres binaires jumeaux non étiquetés et les permutations de Baxter [DG94]. En outre, étant donné que leur algorithme retourne toujours une permutation de Baxter, notre algorithme retourne une permutation de Baxter sur l'entrée d'un couple d'arbres binaires de recherche jumeaux étiqueté par une permutation.

Algorithme : EXTRAIREBAXTER.

Entrée : un couple d'arbres binaires de recherche jumeaux (T_L, T_R) .

Sortie : un mot figurant dans la classe de Baxter encodée par (T_L, T_R) .

1. Soit $u := \epsilon$ le mot vide.
2. Tant que $T_L \neq \perp$ et $T_R \neq \perp$:
 - (a) Soit a l'étiquette de la racine de T_L .
 - (b) Soit i l'indice de la racine de T_L .
 - (c) Poser $u := u \cdot a$.
 - (d) Soit A (resp. B) le sous-arbre gauche (resp. droit) de T_L .
 - (e) Si le i^{e} nœud de T_R est un fils gauche dans T_R :
 - i. alors, poser $T_L := A / B$.
 - ii. Sinon, poser $T_L := A \setminus B$.

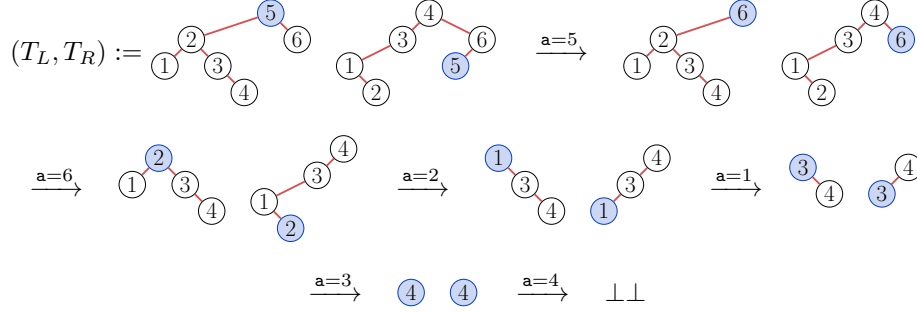


FIGURE 4.6 – Une exécution de l’algorithme EXTRAIREBAXTER sur le couple d’arbres binaires de recherche jumeaux (T_L, T_R) étiqueté par une permutation. La permutation de Baxter calculée est 562134.

- (f) Supprimer le i^{e} nœud de T_R .
 3. Retourner u .

Fin.

La figure 4.6 montre une exécution de cet algorithme.

Les résultats de Dulucq et Guibert permettent immédiatement de montrer que l’exécution de EXTRAIREBAXTER se termine pour toute entrée. Il reste seulement à montrer que le mot calculé appartient bien à la classe de Baxter encodée par le couple d’arbres binaires de recherche jumeaux en entrée. Pour cela, nous avons besoin du lemme suivant.

Lemme 4.3.10. *Soit (T_L, T_R) un couple d’arbres binaires jumeaux non vide. Si la racine de T_L est le i^{e} nœud de T_L , alors le i^{e} nœud de T_R n’a pas de fils.*

Démonstration. Supposons que $T_L = A \wedge B$. Notons que si A et B sont vides, T_L et T_R sont réduits à un nœud, et le lemme est vérifié.

Si $A \neq \perp$, supposons par l’absurde que le i^{e} nœud de T_R possède un sous-arbre gauche non vide. Ceci implique que la i^{e} feuille de T_R n’est pas attachée à son i^{e} nœud. Ainsi, par le lemme 4.3.2, la i^{e} feuille de T_R est attachée à son $(i-1)^{\text{e}}$ nœud et est orientée vers la droite. Dans T_L , la i^{e} feuille ne peut pas être attachée à son i^{e} nœud car $A \neq \perp$. Par conséquent, par le lemme 4.3.2, la i^{e} feuille de T_L est également attachée à son $(i-1)^{\text{e}}$ nœud et est orientée vers la droite. Comme T possède au moins i nœuds, il possède au moins $i+1$ feuilles, ce qui implique que la i^{e} feuille n’est pas la feuille la plus à droite de T_L et T_R , et ainsi, (T_L, T_R) n’est pas un couple d’arbres binaires jumeaux, ce qui contredit l’hypothèse.

Supposons maintenant que le i^{e} nœud de T_R possède un sous-arbre droit non vide. Ceci implique que la $(i+1)^{\text{e}}$ feuille de T_R n’est pas attachée à son i^{e} nœud, et donc, par le lemme 4.3.2, la $(i+1)^{\text{e}}$ feuille de T_R est orientée vers la gauche. De plus, comme le i^{e} nœud de T_R possède un sous-arbre droit non vide et que le i^{e} nœud de T_L est sa racine, le i^{e} nœud de T_L possède également un sous-arbre droit non vide. Ceci implique que la $(i+1)^{\text{e}}$ feuille de T_L n’est pas attachée à son i^{e} nœud, et donc, par le lemme 4.3.2, la $(i+1)^{\text{e}}$ feuille de T_R est aussi orientée vers la gauche. Ceci contredit le fait que (T_L, T_R) est un couple d’arbres binaires jumeaux, et implique que le i^{e} nœud de T_R n’a pas de fils.

Le cas $B \neq \perp$ se traite de la même manière. □

Proposition 4.3.11. *Pour tout couple d’arbres binaires de recherche jumeaux (T_L, T_R) , l’algorithme EXTRAIREBAXTER calcule un mot qui appartient à la classe de Baxter encodée par (T_L, T_R) . De plus, si (T_L, T_R) est étiqueté par une permutation, le mot retourné est une permutation de Baxter.*

Démonstration. Montrons par récurrence sur la taille n de (T_L, T_R) que si (T_L, T_R) est un couple d'arbres binaires de recherche jumeaux, alors EXTRAIREBAXTER retourne un mot qui est à la fois une extension linéaire de $\Delta(T_L)$ et de $\nabla(T_R)$, i.e., comme expliqué précédemment, un mot appartenant à la classe de Baxter encodée par (T_L, T_R) . Cette propriété se vérifie pour $n \leq 1$. Supposons maintenant que $T_L = A \wedge_a B$. Comme T_L et T_R sont des arbres binaires de recherche et sont étiquetés par un même mot, leur i^e nœud possède la même étiquette. Ainsi, par le lemme 4.3.10, il existe dans T_R un nœud x sans fils et étiqueté par a . De plus, la canopée de T_L est de la forme $v01w$ où $v := \text{cnp}(A)$, $w := \text{cnp}(B)$, et la canopée de T_R est de la forme $v'10w'$ où v' (resp. w') est le complémentaire de v (resp. w) étant donné que (T_L, T_R) est un couple d'arbres binaires jumeaux. Nous avons maintenant deux cas.

Si x est un fils gauche dans T_R , l'algorithme retourne le mot au où u est le mot obtenu en appliquant l'algorithme sur (T'_L, T'_R) où $T'_L = A / B$, et T'_R est obtenu à partir de T_R en lui supprimant le nœud x . En premier lieu, la canopée de T'_L est de la forme $v0w$, et celle de T'_R de la forme $v'1w'$. Comme T'_L et T'_R sont encore des arbres binaires de recherche, (T'_L, T'_R) est bien un couple d'arbres binaires de recherche jumeaux. Et ainsi, par hypothèse de récurrence, le mot u appartient à la classe de Baxter encodée par (T'_L, T'_R) , ce qui montre que au appartient à la classe encodée par (T_L, T_R) puisque au est une extension linéaire à la fois de $\Delta(T_L)$ et de $\nabla(T_R)$. Le cas où x est un fils droit dans T_R se traite de la même manière.

Finalement, dans le cas où (T_L, T_R) est étiqueté par une permutation, l'algorithme EXTRAIREBAXTER coïncide avec un algorithme de Dulucq et Guibert [DG94] pour lequel ces derniers ont montré qu'il retourne une permutation de Baxter. \square

La validité de l'algorithme EXTRAIREBAXTER a pour conséquence les deux résultats suivants :

Théorème 4.3.12. *Pour tout $n \geq 0$, il existe une bijection entre l'ensemble des classes d'équivalence de Baxter et l'ensemble des couples d'arbres binaires de recherche jumeaux de taille n .*

Démonstration. Par la proposition 4.3.4 et le théorème 4.3.5, l'algorithme du \mathbb{P} -symbole définit une injection entre l'ensemble des classes de Baxter et l'ensemble des couples d'arbres binaires de recherche jumeaux. De plus, par la proposition 4.3.11, l'algorithme EXTRAIREBAXTER exhibe une surjection entre ces deux ensembles. Ils sont par conséquent en bijection. \square

Le théorème 4.3.12 implique en particulier que les classes de Baxter de permutations de taille n sont en bijection avec l'ensemble des couples d'arbres binaires de recherche jumeaux étiquetés par une permutation — ou de manière équivalente, avec l'ensemble des couples d'arbres binaires jumeaux non étiquetés.

Théorème 4.3.13. *Pour tout $n \geq 0$, tout classe d'équivalence de $\mathfrak{S}_n / \equiv_B$ contient exactement une permutation de Baxter.*

Démonstration. Soit C une classe de $\mathfrak{S}_n / \equiv_B$. Par le théorème 4.3.12, C peut être encodée par un couple d'arbres binaires jumeaux J . Par la proposition 4.3.11, l'algorithme EXTRAIREBAXTER calcule une permutation appartenant à la classe de Baxter encodée par J , impliquant que chaque classe de Baxter contient au minimum une permutation de Baxter. Le théorème suit du fait qu'il y a autant de permutations de Baxter que de couples d'arbres binaires jumeaux. \square

Extraction des mots minimaux et maximaux

Reading définit dans [Rea05] les *permutations de Baxter tordues*, qui sont les permutations qui évitent les motifs généralisés $2-41-3$ et $3-41-2$. Ces permutations sont les permutations minimales pour l'ordre du permutoèdre de nos classes de Baxter de permutations. En effet,

soit σ la permutation minimale dans sa classe de Baxter. Il est alors impossible de réaliser une réécriture de la forme

$$\mathbf{x} u \mathbf{d} \mathbf{a} v \mathbf{y} \rightarrow \mathbf{x} u \mathbf{a} \mathbf{d} v \mathbf{y}, \quad (4.3.8)$$

où $\mathbf{a} < \mathbf{x}, \mathbf{y} < \mathbf{d}$ et $u, v \in A^*$. Ainsi, σ évite les motifs $2 - 41 - 3$ et $3 - 41 - 2$. Réciproquement, si σ est une permutation de Baxter tordue, elle évite les motifs $2 - 41 - 3$ et $3 - 41 - 2$ si bien qu'il est impossible d'appliquer la règle de réécriture \rightarrow dans σ , et par la proposition 4.2.9 et le lemme 4.2.10, σ est minimale de sa classe. Ceci implique que les permutations de Baxter tordues et les permutations de Baxter sont équipotentes puisque par le théorème 4.3.13, il y a exactement une permutation de Baxter par classe de Baxter de permutations, et par la proposition 4.2.9, il y a également exactement une permutation de Baxter tordue. Ceci suggère l'existence d'une bijection envoyant la permutation de Baxter d'une classe vers la permutation de Baxter tordue de cette même classe.

Comme le soulignèrent Law et Reading [LR10], West montra le premier, dans un travail non publié, une bijection entre les permutations de Baxter et les permutations de Baxter tordues, en utilisant la machinerie fournie par les arbres de génération (voir à ce sujet par exemple [Wes90] ou [BM03]). Dans notre contexte, ainsi que dans celui de Law et Reading [LR10], cette bijection est celle qui préserve les classes de Baxter. Nous proposons ci-dessous un algorithme pour calculer cette bijection.

Considérons l'algorithme suivant qui permet, étant donné un couple d'arbres binaires de recherche jumeaux (T_L, T_R) , de calculer le plus petit mot pour l'ordre lexicographique de la classe de Baxter qu'il encode.

Algorithme : EXTRAIREMIN.

Entrée : un couple d'arbres binaires de recherche jumeaux (T_L, T_R) .

Sortie : le plus petit mot pour l'ordre lexicographique de la classe de Baxter encodée par (T_L, T_R) .

1. Soit $u := \epsilon$ le mot vide.
2. Soit $F := T_L$ une forêt ordonnée d'arbres binaires. initialement composée de l'unique arbre T_L .
3. Tant que F n'est pas vide et que $T_R \neq \perp$:
 - (a) Soit i le plus petit entier tel que le i^{e} nœud de F est une racine et le i^{e} nœud de T_R n'a pas de fils.
 - (b) Soit \mathbf{a} l'étiquette du i^{e} nœud de T_L .
 - (c) Poser $u := u \cdot \mathbf{a}$.
 - (d) Supprimer le i^{e} nœud de F et le i^{e} nœud de T_R .
4. Retourner u .

Fin.

Notons que si l'on choisit dans l'instruction (3a) le plus grand entier i possible au lieu du plus petit, le précédent algorithme calcule le plus grand mot pour l'ordre lexicographique de la classe de Baxter encodée par (T_L, T_R) . Nous appelons cette variante EXTRAIREMAX.

La figure 4.7 montre une exécution de cet algorithme.

Proposition 4.3.14. *Pour tout couple d'arbres binaires de recherche jumeaux (T_L, T_R) , l'algorithme EXTRAIREMIN (resp. EXTRAIREMAX) calcule le plus petit (resp. le plus grand) mot pour l'ordre lexicographique de la classe d'équivalence de Baxter encodée par (T_L, T_R) . De plus, si (T_L, T_R) est étiqueté par une permutation, le mot retourné est la plus petite (resp. la plus grande) permutation de sa classe de Baxter pour l'ordre du permutoèdre.*

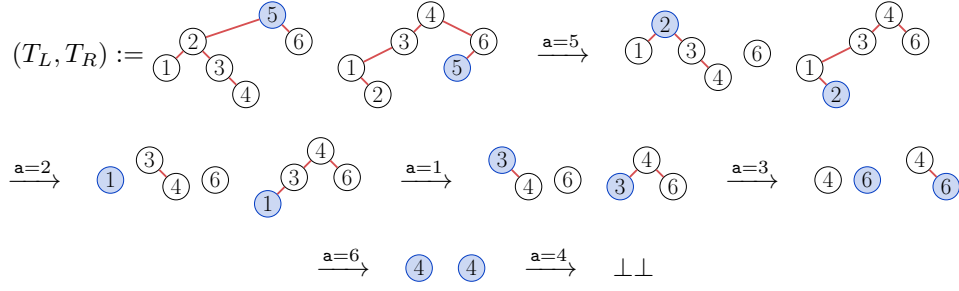


FIGURE 4.7 – Une exécution de l’algorithme EXTRAIREMIN sur le couple d’arbres binaires de recherche jumeaux (T_L, T_R) étiqueté par une permutation. La permutation calculée est 521364 et est le plus petit élément de sa classe de Baxter.

Démonstration. La sortie u de l’algorithme EXTRAIREMIN est clairement à la fois une extension linéaire de $\Delta(T_L)$ et de $\nabla(T_R)$. Ceci implique que u appartient à la classe de Baxter encodée par (T_L, T_R) . De plus, l’exécution de cet algorithme se termine puisque par le théorème 4.3.13, pour tout couple d’arbres binaires de recherche jumeaux, il existe au moins un mot qui est à la fois une extension linéaire de $\Delta(T_L)$ et de $\nabla(T_R)$. La minimalité (resp. maximalité) de la permutation calculée provient de fait qu’à chaque étape, le nœud qui possède la plus petite (resp. la plus grande) étiquette est choisi.

Finalement, étant donné que l’ordre lexicographique sur les permutations est une extension linéaire de l’ordre du permutoèdre, et que d’après la proposition 4.2.9, les classes d’équivalence de Baxter de permutations sont des intervalles du permutoèdre, l’algorithme EXTRAIREMIN (resp. EXTRAIREMAX) retourne la plus petite (resp. la plus grande) permutation de sa classe pour l’ordre du permutoèdre. \square

D’après la proposition 4.3.14, et en utilisant notre analogue de l’algorithme d’insertion de Robinson-Schensted, nous pouvons calculer la bijection entre les permutations de Baxter et les permutations de Baxter tordues de la façon suivante. Si σ est une permutation de Baxter, en appliquant EXTRAIREMIN sur $\mathbb{P}(\sigma)$, nous obtenons la permutation de Baxter tordue en bijection avec σ . Réciproquement, si σ est une permutation de Baxter tordue, en appliquant EXTRAIREBAXTER sur $\mathbb{P}(\sigma)$, nous obtenons la permutation de Baxter en bijection avec σ .

De manière similaire, nous pouvons calculer simplement une bijection entre les permutations de Baxter et les permutations qui évitent les motifs $2-14-3$ et $3-14-2$. Ces permutations sont les plus grands éléments de leur classe de Baxter, et la bijection est analogue à celle que nous venons de décrire en utilisant cette fois-ci l’algorithme EXTRAIREMAX.

4.3.4 Version itérative de l’algorithme d’insertion

Dans cette partie, nous proposons une version itérative du \mathbb{P} -symbole de Baxter, ou en d’autres termes, un algorithme d’insertion \leftarrow tel que pour tout mot u vérifiant $\mathbb{P}(u) = (T_L, T_R)$, l’insertion d’une lettre a dans le couple d’arbres binaires de recherche jumeaux (T_L, T_R) vérifie $(T_L, T_R) \leftarrow a = \mathbb{P}(ua)$.

Ceci — additionnellement au fait d’être en accord avec l’algorithme classique d’insertion de Robinson-Schensted qui lui-même est itératif — donne un moyen de calculer dans le monoïde de Baxter. En effet, rappelons que d’après le théorème 4.3.5, toute classe de Baxter peut s’encoder par un couple d’arbres binaires de recherche jumeaux. Ainsi, pour concaténer deux classes de Baxter dans le monoïde de Baxter, ou de manière équivalente, deux couples d’arbres binaires de recherche jumeaux (T_L, T_R) et (T'_L, T'_R) , il suffit de calculer un mot u' appartenant à la classe

encodée par (T'_L, T'_R) , de manière indifférente avec l'algorithme EXTRAIREBAXTER, EXTRAIREMIN, ou EXTRAIREMAX puis d'insérer les lettres de u' de la gauche vers la droite dans (T_L, T_R) .

Nous avons besoin de quelques définitions avant de poser cet algorithme. Soit T un arbre binaire de recherche droit et b une lettre. La *restriction de T aux valeurs inférieures à b* , noté $T_{\leq b}$, est l'arbre binaire de recherche droit composé uniquement des nœuds x de T étiquetés par des lettres a vérifiant $a \leq b$, et tel que pour tous nœuds x et y de $T_{\leq b}$, si x est un ancêtre de y dans $T_{\leq b}$, alors x est aussi un ancêtre de y dans T . De manière analogue, on définit la *restriction de T aux valeurs strictement supérieures à b* , noté $T_{> b}$. La figure 4.8 illustre ces définitions.

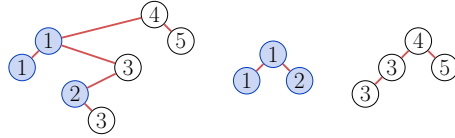


FIGURE 4.8 – Un arbre binaire de recherche droit T et les arbres binaires $T_{\leq 2}$ et $T_{> 2}$.

L'*insertion en racine* d'une lettre a dans T consiste à modifier T de sorte que sa racine soit un nouveau nœud étiqueté par a , son sous-arbre gauche $T_{\leq a}$, et son sous-arbre droit $T_{> a}$.

Nous pouvons maintenant énoncer les versions itératives des \mathbb{P} et \mathbb{Q} -symboles de Baxter :

Définition 4.3.15. Soit (T_L, T_R) un couple d'arbres binaires de recherche jumeaux et a une lettre. L'insertion de a dans (T_L, T_R) consiste à insérer en feuille a dans T_L , et à insérer en racine a dans T_R . Le \mathbb{P} -symbole itératif de Baxter d'un mot u est le couple d'arbres binaires obtenu en insérant successivement les lettres de u de gauche à droite dans le couple (\perp, \perp) . Le \mathbb{Q} -symbole itératif de Baxter de u est le couple d'arbres binaires jumeaux (S_L, S_R) où les nœuds de S_L (resp. S_R) sont étiquetés par leur moment de création dans T_L (resp. T_R).

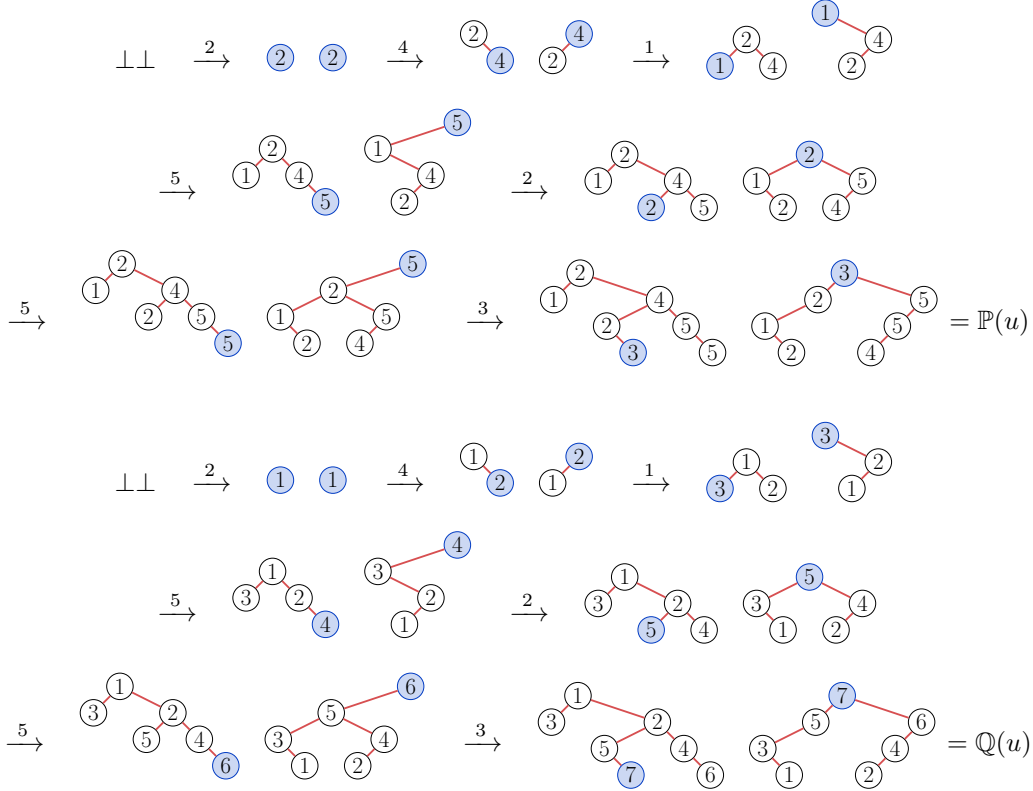
La figure 4.9 illustre, étape par étape, un calcul des \mathbb{P} et \mathbb{Q} -symboles itératifs de Baxter d'un mot.

Pour montrer que le \mathbb{P} -symbole itératif fourni par la définition 4.3.15 construit le même couple d'arbres binaires de recherche jumeaux que le \mathbb{P} -symbole de la définition 4.3.1, montrons le lemme qui suit.

Lemme 4.3.16. Soit u un mot, T l'arbre binaire de recherche droit obtenu en insérant en racine les lettres de u de la gauche vers la droite, et T' l'arbre binaire de recherche droit obtenu en insérant en feuille les lettres de u de la droite vers la gauche. Alors, $T = T'$.

Démonstration. Procédons par récurrence sur $|u|$. Si $u = \epsilon$, le lemme est vérifié puisque $T = T' = \perp$. Sinon, supposons que $u = va$ avec $a \in A$. Soit S l'arbre binaire de recherche droit obtenu par insertion en racine des lettres de v , lu de la gauche vers la droite. Par hypothèse de récurrence, S est aussi l'arbre binaire de recherche droit obtenu par des insertions en feuille des lettres de v lu de la droite vers la gauche. Par définition, T vérifie $T = S_{\leq a} \wedge_a S_{> a}$, et T' est de la forme $T' = L' \wedge_a R'$ où le sous-arbre L' dépend uniquement du sous-mot $v_{\leq a} := v_{] -\infty, a]}$ et le sous-arbre R' dépend uniquement du sous-mot $v_{> a} := v_{] a, +\infty[}$. Ainsi, par hypothèse de récurrence, $L' = S_{\leq a}$ et $R' = S_{> a}$, montrant $T = T'$. \square

Proposition 4.3.17. Pour tout mot u , le \mathbb{P} -symbole de Baxter de u et le \mathbb{P} -symbole itératif de u sont égaux.

FIGURE 4.9 – Les étapes de calcul du \mathbb{P} et du \mathbb{Q} -symbole de Baxter de $u := 2415253$.

Démonstration. Soit (T_L, T_R) le \mathbb{P} -symbole de u et (T'_L, T'_R) le \mathbb{P} -symbole itératif de u . Par définition de ces deux algorithmes d'insertion, on a $T_L = T'_L$. De plus, T_R est obtenu par des insertions en feuille des lettres de u lues de la droite vers la gauche, et T'_R est obtenu par des insertions en racine des lettres de u de la gauche vers la droite. Par le lemme 4.3.16, $T_R = T'_R$. \square

La proposition 4.3.17 implique aussi que le \mathbb{Q} -symbole de Baxter et le \mathbb{Q} -symbole itératif de Baxter de tout mot u sont les mêmes.

4.4 Un treillis sur les couples d'arbres binaires jumeaux

Lorsqu'une relation d'équivalence sur les éléments d'un treillis possède certaines propriétés, celle-ci permet de définir un treillis quotient. Nous rappelons la définition des congruences de treillis et ensuite, montrons que la relation de Baxter, qui est une congruence de monoïde, est aussi une congruence de treillis du permutoèdre. De cela, nous en déduisons une structure de treillis sur l'ensemble des couples d'arbres binaires jumeaux. Nous caractérisons ses relations de couverture puis introduisons un nouvel objet combinatoire en bijection avec les permutations de Baxter, à savoir les diagrammes de Tamari doubles, qui se révèlent être adaptés pour tester si deux éléments sont comparables dans le treillis des couples arbres binaires jumeaux.

Remarquons que notre treillis des couples d'arbres binaires jumeaux est isomorphe au treillis obtenu par Reading en [Rea05] lorsqu'il considère le quotient du permutoèdre par sa congruence de treillis. Une étude complémentaire de ce treillis — dont les objets sont vus comme des quadrangulations — fut réalisée en [LR10].

4.4.1 Congruences de treillis

Si I est un intervalle d'un poset, nous notons $I\uparrow$ (resp. $I\downarrow$) le plus petit (resp. le plus grand) élément de I .

Rappelons la définition suivante introduite en [CS98], que nous adaptons ici dans le contexte des treillis finis.

Définition 4.4.1. Soit (L, \leq) un treillis fini et \equiv une relation d'équivalence sur L . L'équivalence \equiv est également une congruence de treillis si les trois conditions suivantes sont vérifiées :

- (i) toute classe d'équivalence de \equiv est un intervalle de L ;
- (ii) pour tous $x, y \in L$, si $x \leq y$, alors $\hat{x}\uparrow \leq \hat{y}\uparrow$, où \hat{x} et \hat{y} sont les classes d'équivalence respectives de x et de y ;
- (iii) pour tous $x, y \in L$, si $x \leq y$, alors $\hat{x}\downarrow \leq \hat{y}\downarrow$, où \hat{x} et \hat{y} sont les classes d'équivalence respectives de x et de y .

Chajda et Snášel [CS98] ont montré qu'une congruence de treillis permet de construire un poset de la manière suivante :

Définition 4.4.2. Soit (L, \leq) un treillis fini et \equiv une congruence de treillis de L . Alors, le poset quotient de L par \equiv est l'ensemble L/\equiv muni de la relation d'ordre \leq_{\equiv} définie pour toutes classes d'équivalence \hat{x} et \hat{y} de \equiv par

$$\hat{x} \leq_{\equiv} \hat{y} \quad \text{si et seulement si} \quad \text{il existe } x \in \hat{x} \text{ et } y \in \hat{y} \text{ tels que } x \leq y. \quad (4.4.1)$$

Il s'avère de plus que le quotient d'un treillis est aussi un treillis (voir par exemple [Rea04] et [Rea05]), ce qui rend cette définition particulièrement agréable puisqu'elle permet d'obtenir de nouveaux treillis à partir d'anciens. Comme nous l'avons déjà mentionné, il est naturel d'observer le comportement d'une congruence de monoïde d'un bon monoïde en la restreignant aux permutations, et, par le fait que le permutoèdre munit l'ensemble des permutations d'une structure de treillis, il est également naturel de se demander si la congruence de monoïde dont il est question est également une congruence de treillis. Si la réponse est positive, on obtient ainsi gratuitement une structure de treillis sur les objets combinatoires qui indexent les classes de permutations. C'est précisément ce que nous montrons dans le paragraphe suivant au sujet de la congruence de Baxter.

4.4.2 Le treillis des couples d'arbres binaires jumeaux

Si σ est une permutation, on note $\sigma\uparrow$ (resp. $\sigma\downarrow$) le plus petit (resp. le plus grand) élément de sa classe de Baxter. Rappelons que l'existence de ces deux éléments est assurée par la proposition 4.2.9.

Une congruence du permutoèdre

Théorème 4.4.3. La relation d'équivalence de Baxter est une congruence de treillis du permutoèdre.

Démonstration. D'après la proposition 4.2.9, toute classe de Baxter de permutations est bien un intervalle du permutoèdre. Il reste donc à montrer les deux derniers points de la définition 4.4.1.

Soient σ et ν deux permutations telles que $\sigma \leq_P \nu$. Montrons alors que $\sigma\downarrow \leq_P \nu\downarrow$. Il suffit de montrer cette inégalité uniquement lorsque $\nu = \sigma \cdot s_i$ où s_i est une transposition élémentaire et i n'est pas une descente de σ . Si $\sigma = \sigma\downarrow$, alors $\sigma\downarrow \leq_P \nu \leq_P \nu\downarrow$ et la propriété est établie. Autrement, par le lemme 4.2.10, il existe une transposition élémentaire s_j et une permutation

π tels que π et σ sont adjacents pour \xleftrightarrow{B} , $\pi = \sigma \cdot s_j$, et $\sigma \leq_P \pi$. Il reste à démontrer qu'il existe une permutation μ telle que $\nu \equiv_B \mu$ et $\pi \leq_P \mu$. En effet, ceci mène à montrer, en réappliquant itérativement ce raisonnement, que $\sigma \downarrow$ est plus petite pour l'ordre du permutoèdre qu'une permutation appartenant à la classe de Baxter de ν , et ainsi, par transitivité, que $\sigma \downarrow \leq_P \nu \downarrow$. Nous avons quatre cas à explorer suivant les indices i et j .

Cas 1. Si $j \leq i - 2$, σ est de la forme $\sigma = uabvcdw$ où a (resp. c) est la j^e (resp. i^e) lettre de σ et $a < b$ et $c < d$ puisque i et j ne sont pas des descentes de σ . Nous avons $\nu = uabvcdw$ et $\nu \cdot s_j = ubavcdw =: \mu$. De plus, comme $\pi \xleftrightarrow{B} \sigma$, il existe des lettres $x \in \text{Alph}(u)$ et $y \in \text{Alph}(vcdw)$ telles que $a < x, y < b$. Ainsi, $\mu \xleftrightarrow{B} \nu$. Finalement, puisque $\pi = uabvcdw$, $\pi \leq_P \mu$, de sorte que μ est la permutation recherchée.

Cas 2. le cas $j \geq i + 2$ est analogue au précédent.

Cas 3. si $j = i + 1$, σ est de la forme $\sigma = uabcv$ où a est la i^e lettre de σ et $a < b < c$ puisque i et j ne sont pas des descentes de σ . Comme $\sigma \xleftrightarrow{B} \pi$, il existe des lettres $x \in \text{Alph}(u)$ et $y \in \text{Alph}(v)$ telles que $b < x, y < c$. Ainsi, puisque $\nu = ubacv$ et $a < b < x, y < c$, on a $\nu \cdot s_j = ubcav \xleftrightarrow{B} \nu$. De plus, $\nu \cdot s_j \cdot s_i = ucba v =: \mu$ et $\nu \cdot s_j \xleftrightarrow{B} \nu \cdot s_j \cdot s_i$ puisque $b < x, y < c$ et donc, $\mu \equiv_B \nu$. Finalement, comme $\pi = uabcv$, on $\pi \leq_P \mu$, et ainsi μ est la permutation recherchée.

Cas 4. le cas $j = i - 1$ est analogue au précédent.

Le cas $\sigma \uparrow \leq_P \nu \uparrow$ se traite de la même manière. □

Le treillis de Baxter

Rappelons que par le théorème 4.3.12, toute classe de Baxter de permutations de taille n peut s'encoder par un couple d'arbres binaires jumeaux. De ce fait, et par la définition 4.4.2, le quotient du permutoèdre d'ordre n par la relation de Baxter est le treillis de la forme (\mathcal{J}_n, \leq_B) où la relation d'ordre \leq_B vérifie, pour tout $J_0, J_1 \in \mathcal{J}_n$,

$$J_0 \leq_B J_1 \quad \text{si et seulement si} \quad \begin{array}{l} \text{il existe } \sigma, \nu \in \mathfrak{S}_n \text{ tels que} \\ \sigma \leq_P \nu, \mathbb{P}(\sigma) = J_0 \text{ et } \mathbb{P}(\nu) = J_1. \end{array} \quad (4.4.2)$$

Nous appelons ce treillis le *treillis de Baxter*. La figure 4.10 montre les classes de Baxter dans le permutoèdre d'ordre 4.

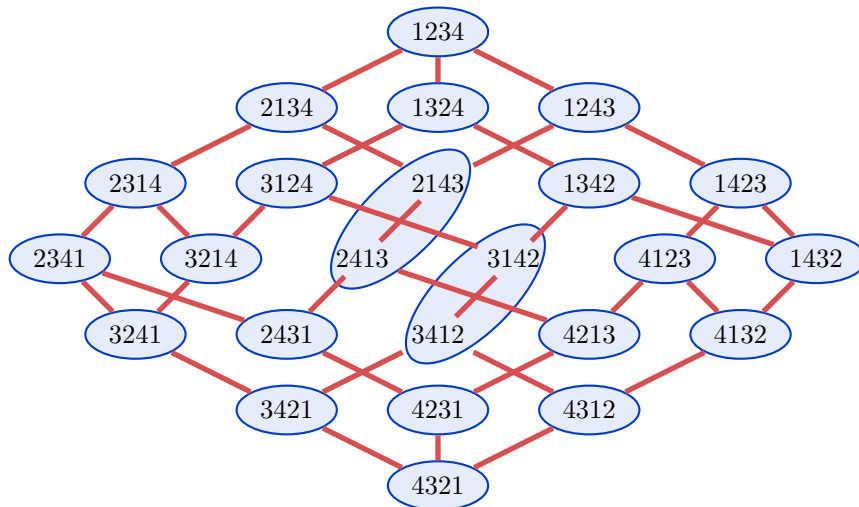


FIGURE 4.10 – Le permutoèdre d'ordre 4 découpé en classes d'équivalence de Baxter.

Relations de couvertures

Décrivons maintenant ses relations de couverture. Considérons pour cela une classe $\hat{\sigma}$ de Baxter de permutations encodée par le couple d'arbres binaires jumeaux (T_L, T_R) , et σ le plus grand élément de $\hat{\sigma}$. Pour toute descente en position i de σ , la permutation $\sigma \cdot s_i$ n'est plus dans $\hat{\sigma}$, et, par définition du treillis de Baxter, le couple d'arbres binaires jumeaux $\mathbb{P}(\sigma \cdot s_i) =: (T'_L, T'_R)$ couvre (T_L, T_R) . Les permutations σ et $\sigma \cdot s_i$ sont de la forme

$$\sigma = u \mathbf{ad} v \quad \text{et} \quad \sigma \cdot s_i = u \mathbf{da} v, \quad (4.4.3)$$

où $\mathbf{a} < \mathbf{d}$. Il y a trois cas différents suivant si le facteur u ou v possède une lettre intermédiaire en valeur entre \mathbf{a} et \mathbf{d} . En se basant sur le fait que la relation sylvestre est une congruence du permutoèdre, que celle-ci permet de définir le treillis de Tamari (voir la démonstration dans [HNT05]), et que les couvertures de ce dernier sont des rotations (voir la définition 1.3.1 du chapitre 1), nous trouvons les couvertures suivantes dans le treillis de Baxter :

1. s'il existe une lettre y dans v telle que $\mathbf{a} < y < \mathbf{d}$, alors $T'_R = T_R$ et T'_L est obtenu en réalisant une rotation gauche dans T_L qui ne modifie pas sa canopée ;
2. s'il existe une lettre x dans u telle que $\mathbf{a} < x < \mathbf{d}$, alors $T'_L = T_L$ et T'_R est obtenu en réalisant une rotation droite dans T_R qui ne modifie pas sa canopée ;
3. si aucune lettre de u ni de v n'est comprise en valeur entre \mathbf{a} et \mathbf{d} , alors T'_L (resp. T'_R) est obtenu en réalisant une rotation gauche (resp. rotation droite) dans T_L (resp. T_R) qui modifie sa canopée.

Ainsi, d'après cette caractérisation des relations de couverture du treillis de Baxter et la définition de l'ordre de Tamari (voir la définition 1.3.2 du chapitre 1), nous avons, pour tous couples d'arbres binaires jumeaux (T_L, T_R) et (T'_L, T'_R) ,

$$(T_L, T_R) \leqslant_{\mathbf{B}} (T'_L, T'_R) \quad \text{si et seulement si} \quad T'_L \leqslant_{\mathbf{T}} T_L \text{ et } T_R \leqslant_{\mathbf{T}} T'_R. \quad (4.4.4)$$

Notons qu'une rotation droite en y réalisée dans un arbre binaire T modifie sa canopée si et seulement si le sous-arbre droit B du fils gauche x de y est vide (voir la figure 1.4 du chapitre 1). Similairement, une rotation gauche en y modifie la canopée de T si et seulement si le sous-arbre gauche de B de y est vide. De plus, si y est le i^{e} nœud de T , par le lemme 4.3.2, nous pouvons observer que B est aussi la i^{e} feuille de T . Par conséquent, une rotation droite (resp. gauche) en y modifie l'orientation de la i^{e} feuille de T de la droite vers la gauche (resp. de la gauche vers la droite).

4.4.3 Diagrammes de Tamari doubles

Décider si deux couples d'arbres binaires jumeaux sont comparables dans le treillis de Baxter n'est pas chose évidente. Le but de ce paragraphe est d'introduire un nouvel objet combinatoire en bijection avec les couples d'arbres binaires jumeaux. Celui-ci permet de réaliser de manière transparente les comparaisons dans le treillis de Baxter.

Diagrammes de Tamari

Nous nous basons sur le fait qu'il existe un moyen de comparer facilement deux arbres binaires dans le treillis de Tamari en utilisant l'objet combinatoire suivant :

Définition 4.4.4. *Un diagramme de Tamari est un mot u sur l'alphabet \mathbb{N} qui respecte les deux conditions suivantes :*

- (i) $0 \leqslant u_i \leqslant n - i$ pour tout $i \in [n]$;

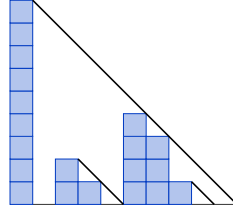


FIGURE 4.11 – Le diagramme de Tamari 9021043100 dans sa représentation graphique et ses diagonales.

(ii) $u_{i+j} \leq u_i - j$ pour tout $i \in [n]$ et $j \in [u_i]$.

La taille d'un diagramme de Tamari est son nombre de lettres.

Un diagramme de Tamari se représente avantageusement sous une forme graphique, en représentant chaque lettre par des colonnes dont les hauteurs dépendent des valeurs des lettres. Vérifier la condition (ii) devient alors très simple puisqu'il suffit de tracer du sommet de chaque colonne une diagonale de pente $-\frac{1}{2}$ comme indiqué dans la figure 4.11, et de s'assurer qu'aucune colonne n'en dépasse. De même, on vérifie la condition (i) en s'assurant qu'aucune diagonale ne tombe à droite du diagramme.

Pallo [Pal86] montra que l'ensemble des diagrammes de Tamari de taille n est en bijection avec l'ensemble des arbres binaires à n nœuds. Cette bijection se décrit de la manière suivante. On associe à un arbre binaire T le diagramme de Tamari $\text{dt}(T)$ obtenu en étiquetant chaque nœud x de T par le nombre de nœuds du sous-arbre droit de x et considérant le mot obtenu par sa lecture infixe. Réciproquement, si u est un diagramme de Tamari de taille n , on lui associe un arbre binaire $\text{dt}^{-1}(u)$ récursivement de la manière suivante :

$$\text{dt}^{-1}(u) := \begin{cases} \perp & \text{si } u = \epsilon, \\ \text{dt}^{-1}(u_1 \dots u_{i-1}) \wedge \text{dt}^{-1}(u_{i+1} \dots u_n) & \text{où } i := \min\{i : u_i = n - i\} \text{ sinon.} \end{cases} \quad (4.4.5)$$

Les diagrammes de Tamari sont de bonnes manières de voir les sommets du treillis de Tamari puisque l'ordre s'y lit naturellement [Knu04]. En effet, si T et T' sont deux arbres binaires à n nœuds, nous avons

$$T \leq_T T' \quad \text{si et seulement si} \quad \text{dt}(T)_i \leq \text{dt}(T')_i \text{ pour tout } i \in [n]. \quad (4.4.6)$$

Diagrammes de Tamari doubles

Si u est un mot de \mathbb{Z}^* , nous notons par $u \uparrow$ l'image de u par le morphisme de monoïde qui envoie toute lettre $a \in \mathbb{Z}$ vers $\max\{0, a\}$. De même, on note $u \downarrow$ l'image de u par le morphisme de monoïde qui envoie toute lettre $a \in \mathbb{Z}$ vers $-\min\{0, a\}$. Par exemple, pour $u := (2, -2, -1, 3, 1, -1, 0)$, nous avons $u \uparrow = (2, 0, 0, 3, 1, 0, 0)$ et $u \downarrow = (0, 2, 1, 0, 0, 1, 0)$.

Définition 4.4.5. Un diagramme de Tamari double $\delta = \delta_1 \dots \delta_n$ est un mot de \mathbb{Z}^* où

- (i) $\delta \uparrow$ et $\delta \downarrow$ sont des diagrammes de Tamari;
- (ii) pour tout $i \in [n-1]$, $\delta_i \neq 0$.

La taille d'un diagramme de Tamari double est son nombre de lettres.

Nous représentons graphiquement un diagramme de Tamari double δ en dessinant, pour chaque lettre δ_i une colonne de hauteur correspondante orientée vers le haut si δ_i est positive

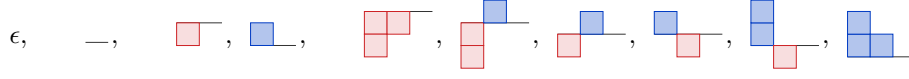


FIGURE 4.12 – Les diagrammes de Tamari doubles de tailles 0, 1, 2 et 3.

ou vers vers le bas sinon. Les premiers diagrammes de Tamari doubles sont représentés suivant cette convention dans la figure 4.12.

Montrons que les diagrammes de Tamari doubles sont des objets de la famille combinatoire de Baxter :

Proposition 4.4.6. *Pour tout $n \geq 0$, l'ensemble des diagrammes de Tamari doubles de taille n est en bijection avec l'ensemble des couples d'arbres binaires jumeaux à n nœuds. De plus, cette bijection se calcule de la manière suivante. Si $J := (T_L, T_R)$ est un couple d'arbres binaires jumeaux, le diagramme de Tamari double $\text{dtd}(J)$ qui lui est associé est défini par*

$$\text{dtd}(J)_i := \begin{cases} u_i & \text{si } u_i \neq 0 \text{ et } v_i = 0, \\ -v_i & \text{si } v_i \neq 0 \text{ et } u_i = 0, \\ 0 & \text{sinon } (i = n), \end{cases} \quad (4.4.7)$$

où $u := \text{dt}(T_R)$ et $v := \text{dt}(T_L)$. Inversement, si δ est un diagramme de Tamari double, on lui associe le couple d'arbres binaires jumeaux

$$\text{dtd}^{-1}(\delta) := (\text{dt}^{-1}(\delta \downarrow), \text{dt}^{-1}(\delta \uparrow)). \quad (4.4.8)$$

Démonstration. Commençons par montrer que (4.4.7) définit bien un diagramme de Tamari double. Soient pour cela $J := (T_L, T_R)$ un couple d'arbres binaires jumeaux de taille n , $u := \text{dt}(T_R)$, $v := \text{dt}(T_L)$ et $\delta := \text{dtd}(J)$. Pour que δ soit bien défini, il faut d'une part que $u_i = v_i = 0$ implique $i = n$, et d'autre part que l'on ait $u_i = 0$ ou $v_i = 0$ pour tout $i \in [n]$. Supposons qu'il existe $i \in [n]$ tel que $u_i = v_i = 0$. Par définition des diagrammes de Tamari, ceci implique que les i^{e} nœuds de T_L et de T_R n'ont pas de fils droit. Ainsi, par le lemme 4.3.2, les $(i+1)^{\text{e}}$ feuilles de T_L et de T_R sont attachées aux i^{e} nœuds de T_L et de T_R et sont orientées vers la droite. Comme J est un couple d'arbres binaires jumeaux, la $(i+1)^{\text{e}}$ feuille est nécessairement la feuille la plus à droite des deux arbres, et ainsi, $i = n$. Pour montrer la seconde condition, supposons par l'absurde qu'il existe $i \in [n]$ tel que $u_i \neq 0$ et $v_i \neq 0$. Par définition des diagrammes de Tamari, ceci implique que les i^{e} nœuds de T_L et de T_R ont tous deux un fils droit non vide. Ainsi, par le lemme 4.3.2, les $(i+1)^{\text{e}}$ feuilles sont nécessairement attachées aux $(i+1)^{\text{e}}$ nœuds de T_L et de T_R , et orientées vers la gauche. Ceci est en contradiction avec le fait que J est un couple d'arbres binaires jumeaux. Le mot δ est ainsi bien défini et est un diagramme de Tamari double.

Inversement, montrons que si δ est un diagramme de Tamari double, le couple $(T_L, T_R) := \text{dtd}^{-1}(\delta)$ est bien un couple d'arbres binaires jumeaux. Par l'absurde, supposons que les i^{e} feuilles de T_L et de T_R sont orientées vers la gauche avec $i \geq 2$. D'après le lemme 4.3.2, ceci implique que les sous-arbres droits des $(i-1)^{\text{e}}$ nœuds de T_L et de T_R sont non vides. Ceci contredit le fait que δ est un diagramme de Tamari double. De même, si l'on suppose par l'absurde que les i^{e} feuilles de T_L et de T_R sont orientées vers la droite avec $i \leq n-1$, alors, par le lemme 4.3.2, ces feuilles sont toutes deux attachées aux $(i-1)^{\text{e}}$ nœuds de T_L et T_R . Ceci contredit également le fait que δ est un diagramme de Tamari double.

On conclut finalement par le fait que l'application dt qui à un arbre binaire associe un diagramme de Tamari est une bijection [Pal86]. \square

Nous avons maintenant le moyen suivant de comparer deux éléments dans le treillis de Baxter :

Proposition 4.4.7. *Soient $J := (T_L, T_R)$ et $J' := (T'_L, T'_R)$ deux couples d'arbres binaires jumeaux de taille n . Nous avons alors, dans le treillis de Baxter,*

$$(T_L, T_R) \leq_B (T'_L, T'_R) \quad \text{si et seulement si} \quad \text{dtd}(J)_i \leq \text{dtd}(J')_i \text{ pour tout } i \in [n]. \quad (4.4.9)$$

Démonstration. Le résultat suit de caractérisation de la relation d'ordre du treillis de Baxter fournie en (4.4.4), du procédé de comparaison de deux arbres binaires dans le treillis de Tamari donné par (4.4.6) en passant par leurs diagrammes de Tamari et de l'expression de la bijection dtd fournie par la proposition 4.4.6 entre les couples d'arbres binaires jumeaux et les diagrammes de Tamari doubles. \square

La figure 4.13 montre un intervalle du treillis de Baxter, où les sommets sont vus comme des couples d'arbres binaires jumeaux, ou de manière équivalente, comme des diagrammes de Tamari doubles.

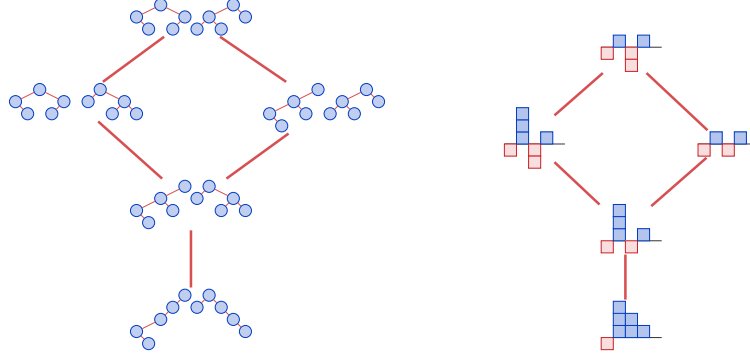


FIGURE 4.13 – Un intervalle du treillis de Baxter d'ordre 5 dont les sommets sont interprétés comme des couples d'arbres binaires jumeaux et comme des diagrammes de Tamari doubles.