Construction du monoïde plaxique, correspondance de Robinson-Schensted-Knuth et théorème d'Erdös-Szekeres

Louis Guo, Florent Maligne, Benjamin Petit

26 juin 2014

Préface

Dans ce rapport, nous nous intéresserons, par l'intermédiaire d'une construction du monoïde plaxique (*Plactic monoid* en anglais), aux permutations, et plus particulièrement à l'étude des sous-suites croissantes de celles- ci. Dans une première partie, nous présenterons, après quelques rappels algébriques élémentaires portant notamment sur les monoïdes, les alphabets et sur le groupe symétrique, la théorie des tableaux de Young, qui sont grossièrement des tableaux de taille arbitraire remplis d'entiers, introduite dès 1900 par Alfred Young et qui trouva de formidables applications à l'étude du groupe symétrique. Nous définirons ensuite l'opération d'insertion, qui permet, par l'intermédiaire d'un algorithme simple, d'insérer une lettre dans un tableau de Young tout en conservant le caractère semi-standard de celui-ci. Nous proposerons également une implémentation simple de ce procédé dans le langage Python.

Par la suite, nous présenterons rapidement les travaux de Donald Knuth, et plus particulièrement la relation d'équivalence portant son nom, qui sera un outil fort utile pour la construction du monoïde plaxique. En effet, nous démontrerons que cette relation d'équivalence portant sur les éléments du monoïde libre engendré par un alphabet fini totalement ordonné est "compatible" avec l'opération d'insertion et préserve la longueur de la plus longue sous-séquence croissante d'un mot, ce qui nous permettra, après démonstration de quelques résultats techniques (lemmes liés notamment aux travaux de Pieri et de Greene) portant sur les sous-suites croissantes d'un mot, de démontrer le fondamental théorème de la section, dû à Lascoux et Schützenberger, qui assure que tout mot du monoïde libre est congru, au sens de Knuth, à un et un seul mot de tableau (c'est-à-dire un mot dont la représentation naturelle est un tableau de Young semi-standard). Ce théorème est au fondement même de l'idée de monoïde plaxique, qui sera défini comme l'ensemble quotient du monoïde libre engendré par un alphabet fini par l'équivalence de Knuth, chaque classe d'équivalence étant représentée par le seul et unique mot de tableau qu'elle contient.

Nous élargirons ensuite cette réprésentation en nous intéressant non plus à un alphabet fini, mais à un couple d'alphabets, en définissant la correspondance de Robinson-Schensted-Knuth (RSK), qui permet d'associer récursivement à tout bi-mot lexicographique (c'est-à-dire à tout mot dont les lettres sont des couples de lettres dans deux alphabets pouvant être distincts, le tout étant ordonné par l'ordre lexicographique) un couple de tableaux semi-standards et, en démontrant le théorème fondamental éponyme, nous prouverons qu'elle définit une bijection entre les deux ensembles sur lesquels elle porte. Après quelques résultats élémentaires de dénombrement de ces ensembles permis par une telle bijection, nous nous intéresserons à une application précise de cette correspondance : l'étude des sous-suites croissantes d'une permutation. En effet, on montrera un dernier théorème dû à Robinson et Schensted, proposant un algorithme lié à un cas particulier de la correspondance RSK (en réalité à l'origine de celle-ci), permettant de déterminer la plus longue sous-séquence croissante d'une permutation. Nous proposerons également une implémentation de cette correspondance, cette fois-ci dans le langage de programmation CamL. La bijection donnée par ce théorème entre le groupe symétrique et l'ensemble des couples de tableaux standards de même forme nous permettra de montrer un résultat fort de dénombrement : l'identité de Frobenius (aussi appelée formule de la somme des carrés), assurant que :

$$\forall n \in \mathbb{N}, \sum_{\lambda \in P_n} (f^\lambda)^2 = n!$$

où P_n est l'ensemble des partitions de n, f^λ est le nombre de tableaux standards de forme λ. Enfin, nous proposerons, grâce au théorème de Robinson-Schensted, une démonstration originale du théorème d'Erdös-Szekeres, assurant l'existence pour une permutation de $pq+1$ d'une sous-séquence croissante de longueur au moins $p+1$ ou d'une sous-séquence décroissante de longueur au moins $q+1$.

Nous nous appuierons notamment sur des articles publiés par Knuth, Lascoux, Schützenberger, Kortchemski, ainsi que sur des notes de cours publiées par François Bergeron, de l'université de Montréal. L'intérêt principal de ce rapport sera de proposer une approche claire et progressive du sujet, ne cherchant cependant pas à être exhaustive, le sujet étant pour le moins extrêmement vaste et amenant de très nombreuses applications.

Table des matières

Première partie

Premières définitions

1 Alphabets et monoïdes

Définition 1. On appelle **monoïde** un couple $(E, \bullet)$ formé d'un ensemble et d'une **loi de composition interne associative** sur cet ensemble, admettant un (nécessairement unique) **élément neutre**, c'est-à-dire d'une application $\bullet : E \times E \to E$ vérifiant :

$$\begin{cases} \forall (a,b,c) \in E^2,\ (a \bullet b) \bullet c = a \bullet (b \bullet c) \\ \exists e \in E, \forall a \in E,\ a \bullet e = e \bullet a = a \end{cases}$$

Définition. On appelle **alphabet fini totalement ordonné** un ensemble fini A muni d'un ordre total $\leq$. Les éléments de cet ensemble seront appelés des lettres et les suites finies $(a_1, ..., a_n)$ de lettres, des mots. On notera, en pratique, $a_1...a_n$ un mot.

On munit l'ensemble des mots, noté $A*$, d'une loi de composition interne appelée concaténation, vérifiant :

$$a_1 a_2 ... a_n \bullet a'_1 a'_2 ... a'_m = a_1 ... a_n a'_1 ... a'_m$$

Muni de cette loi de composition interne, $A*$ possède une structure naturelle de monoïde. En effet, il est clair que cette loi est associative et admet pour élément neutre le mot vide, noté ε.

Définition 2. Soit $(E, \bullet)$ un monoïde et $P \subset E$ une partie de E. On dit que P est une base de E si :

$$\forall a \in E, \exists ! n \in \mathbb{N}, \exists ! (p_1, ..., p_n) \in P^n,\ a = p_1 \bullet ... \bullet p_n$$

Autrement dit, si tout élément de E est décomposable en un unique produit fini d'éléments de P.

Tout monoïde admettant une base sera appelé dit libre. La base est alors unique.

Le monoïde $(A*, \bullet)$ admet une base naturelle formée de l'ensemble des mots de longueur 1 (assimilable à l'ensemble de ses lettres) : $B = \{(a), a \in A\}$.

2 Rappels sur le groupe symétrique

Définition 3. On appelle **groupe symétrique** à n éléments, ou encore **groupe des permutations de** $\{1, ..., n\}$ et on note S_n l'ensemble des bijections de $\{1, ...n\}$ dans lui-même.

On utilisera dans la suite la représentation suivante pour une permutation σ, sous la forme d'un bi-mot de $\{1, ..., n\} \times \{1, ..., n\}$ (notion qui sera définie formellement dans la suite).

$$\sigma : \begin{pmatrix} 1 & 2 & \cdots & \cdots & n \\ \sigma(1) & \sigma(2) & \cdots & \cdots & \sigma(n) \end{pmatrix}$$

Proposition 4. *Comme son nom l'indique, $(S_n, \circ)$ est un groupe (où $\circ$ est la composition des applications). On rappelle que toute permutation est décomposable en produit de cycles, et donc en produit de transpositions. Enfin, on a $\forall n \in \mathbb{N}, Card(S_n) = n!$.*

Démonstration. Ce sont des résultats du chapitre sur les déterminants et le groupe symétrique, au programme de la classe de MPSI. Ils ne seront donc pas démontrés ici. □

Définition 5. Une **sous-suite croissante** d'une permutation $\sigma \in s_n$ est une séquence strictement croissante $(\sigma(a_1), ..., \sigma(a_k))$ vérifiant $\sigma(a_1) < \sigma(a_2) < ... < \sigma(a_k)$ où $a_1 < ... < a_k$. De même, une sous-séquence décroissante d'une permutation $\sigma \in S_n$ est une séquence strictement décroissante $(\sigma(a_1), ..., \sigma(a_k))$ vérifiant $\sigma(a_1) > \sigma(a_2) > ... > \sigma(a_k)$ où $a_1 < ... < a_k$.

Exemple. Considérons la permutation $\sigma : \begin{pmatrix} 1 & 2 & 4 & 5 & 6 & 7 & 8 & 9 \\ 2 & 4 & 3 & 6 & 1 & 9 & 7 & 8 \end{pmatrix}$.

Alors $(2, 3, 6, 7, 8)$, $(2, 4)$, et $(1, 7, 8)$ sont des exemples de sous-suites croissantes de σ.

De même, $(4, 3, 1)$, $(2, 1)$ et $(9, 7, 8)$ sont des exemples de sous-suites décroissantes de σ.

3 Diagrammes et tableaux de Young

3.1 Diagrammes de Young

Définition 6. On appelle **diagramme de Young** un ensemble **fini** $d \subset \mathbb{N} \times \mathbb{N}$. Une **case** de d est simplement un élément de d. On représente un diagramme de Young en représentant ses cases par des carrés de taille 1x1 dans le cadran $\mathbb{R}^+ \times \mathbb{R}^+$.

Exemple. $\{(1,0), (1,1), (2,0), (2,0), (1,3), (3,1)\}$ est un diagramme de Young. Il est représenté graphiquement par :

Définition 7. On dira qu'un diagramme d est un **partage** si :

$$\forall (a,b) \in d, \forall (i,j) \in \mathbb{N}^2, \begin{cases} i \leq a \\ j \leq b \end{cases} \implies (i,j) \in d$$

Autrement dit, un partage est un diagramme contenant toutes les cases qui le délimitent. On notera $\mathbb{P} \subset \mathbb{N} \times \mathbb{N}$ l'ensemble des partages.

On appelle **ligne i**, ou **part i**, pour $i \in \mathbb{N}*$ du partage d l'ensemble $l_i = \{l \in \mathbb{N} * | (l, i-1) \in d\}$.

Il est clair, par conséquence immédiate de la définition, que la longueur des lignes d'un partage est décroissante au sens large (en montant).

On dira qu'un partage est un **partage gauche** lorsqu'il est obtenu par une différence (opération d'exclusion) de deux partages.

Définition. Soit d un diagramme. On appelle **conjugué** de d et on note d^c le diagramme :

$$d^c := \{(i,j) | (j,i) \in d\}$$

Le conjugué d'un partage est toujours un partage.

3.2 Tableaux de Young

Définition 8. Un **tableau de Young** est une application

$$t : \begin{pmatrix} & d \to A & \\ & (i,j) \longmapsto t(i,j) & \end{pmatrix}$$

où d est un partage, et A un alphabet fini. On dit que d est la forme de t et on note $\lambda(t) = d$. Généralement, on prendra $A = \{1, .., n\}$. On représente alors le tableau en remplissant avec les valeurs associées les cases du diagramme de Young. On dira qu'un tableau est **injectif** lorsque la fonction t est injective.

Un tableau sera qualifié de **semi-standard** s'il vérifie une croissance stricte suivant les colonnes et large selon les lignes, c'est-à-dire si :

$$\forall (i,j), \begin{cases} t(i,j) \leq t(i+1,j) \\ t(i,j) < t(i,j+1) \end{cases}$$

On notera $T(\mathbb{A})$ l'ensemble des tableaux de Young semi-standards à valeurs dans $\mathbb{A}$, et on aura $T(\mathbb{A}) \subset \bigcup_{d \in \mathbb{P}} \mathcal{F}(d, \mathbb{A})$, et $T_n(\mathbb{A})$ l'ensemble des tableaux semi-standards à n cases, à valeurs dans $\mathbb{A}$.

Enfin, si t vérifie une croissance stricte à la fois selon les lignes et les colonnes, ie si :

$$\forall (i,j), \begin{cases} t(i,j) < t(i+1,j) \\ t(i,j) < t(i,j+1) \end{cases}$$

Alors on dira que t est un tableau **partiel**. Si t est partiel et à valeurs distinctes dans $\{1, ..., n\}$, on dira que t est un tableau de Young **standard**. On notera T_s l'ensemble des tableaux de Young standards.

Exemple. Le tableau

1	1	
2	1	2
4	3	6
4	3	9

est un tableau de Young quelconque.

Le tableau

8				
5	6	6		
4	5	5	8	
3	3	6	7	11

est semi-standard.

Le tableau

8		
6	7	
3	4	
1	2	5

est standard, tandis que le tableau

8		
6	7	
3	4	
1	2	6

n'est que partiel.

Définition. Soit t un tableau de Young. On note $c_1, ..., c_n$ les cases de t. On appelle **mot de lecture** du tableau t et on note $\omega(t)$ la séquence $\omega(t) = t(c_1)...t(c_n)$.

Exemple. Le mot de lecture du tableau

1	1	
2	1	2
4	3	6
4	3	9

est : 11212436439

4 Opération d'insertion

Définition 9. Soit $m = a_1...a_n$ un mot. On appelle **factorisation en segments croissants maximaux** la factorisation

$$m = \underbrace{(s_{1,1}s_{1,2}\dots s_{1,l_1})}_{s_1}\underbrace{(s_{2,1}\dots s_{2,l_2})}_{s_2}\cdots\underbrace{(s_{k,1}\dots s_{k,l_k})}_{s_k}$$

$$\forall i \in \{1, ..., k\},\ s_{i,1} \leq s_{i,2} \leq \cdots \leq s_{i,l_i}\ et\ \forall j \in \{1, ..., k-1\},\ s_{i,l_i} > s_{i+1,1}$$

Les mots $s_1, ..., s_k$ sont donc des segments croissants au sens large, ceux-ci étant de longueur maximale. Il est clair que, pour $t \in T_n$ un tableau semi-standard, la factorisation du mot de lecture $\omega(t)$ correspond aux différentes lignes de t, en partant cette fois-ci du haut.

Définition. On appelle **mot de tableau** le mot de lecture $\omega(t)$ d'un tableau semi-standard.

Il y a alors bijection entre l'ensemble des mots de tableau de longueur n et T_n, et plus généralement entre l'ensemble des mots de tableau et l'ensemble des tableaux semi-standards.

Définition 10. Soient $m = m_1 \cdots m_n \in A*$ un mot de tableau, $a \in A$ une lettre, et $m = s_1 \cdots s_k$ la factorisation de m en sous-segments croissants maximaux. On définit de cette manière l'opération d'insertion de a dans m, notée $(m \leftarrow a)$:

On pose $s_k = \underbrace{a_1...a_{i-1}}_{\alpha} a_i \underbrace{a_{i+1}...a_l}_{\beta}$ l'écriture du dernier segment croissant de m. α et β sont tels que $a_1 \leq a_2 \leq ... \leq a_{i-1} \leq a < a_i \leq a_{i+1} \leq ... \leq a_l$, ie a_i est le premier élément du segment supérieur strictement à a. Alors :

$$(m \leftarrow a) = \begin{cases} s_1...s_{k-1}(a_1...a_l)x & si\ \alpha = s_k \\ (s_{1...}s_{k-1} \leftarrow a_i)(a_1...a_{i-1}aa_{i+1}...a_l) & sinon \end{cases}$$

On définit alors la **perturbation engendrée par l'insertion de a dans m,** notée $\Pi(m, a)$ comme la suite finie des a_i.

Exemple 11. Insérons, par exemple, 4 dans le mot de tableau $m = 67689448256$.

La factorisation en facteurs croissants maximaux de m est $(67)(689)(448)(256)$. On a donc :

$256 = \underbrace{2}_{\alpha}\underbrace{5}_{\beta}\ 6$ et $5 > 4$. On effectue ainsi l'opération $((67)(689)(448) \leftarrow 5) \bullet (246)$.

En réitérant ce procédé, on a : $((67)(689)(448) \leftarrow 5) = ((67)(489) \leftarrow 8) \bullet (445)$.

Puis $((67)(489) \leftarrow 8) = ((67) \leftarrow 9) \bullet (488)$, et enfin :

$(67 \leftarrow 8) = (678)$, ce qui donne : $(m \leftarrow 4) = (678)(488)(445)(246)$

La perturbation associée est $\Pi(m, 4) = (4, 5, 8, 9)$

On propose en annexe un programme écrit en Python permettant de réaliser cette opération d'insertion de façon récursive.

Lemme 12. *Soient $(a, b) \in A^2$ deux lettres telles que $b \geq a$ et m un mot de tableau. Alors $\Pi(m, a)$ est de longueur supérieure ou égale à $\Pi((m \leftarrow a), b)$.*

Démonstration. Par définition de l'insertion, la suite $\Pi((m \leftarrow a), b)$ est à valeurs supérieures ou égales à $\Pi(m, a)$. Alors, en notant k_a la longueur de $\Pi(m, a)$ ainsi que k_b la longueur de $\Pi((m \leftarrow a), b)$, et en supposant que $k_b \geq k_a$, $\Pi(m, a)_{k_a}$ est le dernier bousculé par l'insertion de a dans m. Donc nécessairement, il est présent tout à droite de sa ligne dans le tableau associé. Alors, $\Pi((m \leftarrow a), b)_{k_a}$ étant supérieur à $\Pi(m, a)_{k_a}$, il est nécessairement ajouté à sa droite et n'entraine plus de perturbation. D'où $k_a = k_b$.

On a donc, dans tous les cas, $k_b \leq k_a$. □

Lemme 13. *Soient $m = m_1...m_n$ un mot de tableau, $m = \omega(t)$, et $a \in A$ une lettre.*

Alors $(m \leftarrow a)$ est un mot de tableau.

Démonstration. Soit $m = m_1...m_n$ un mot de tableau, et $a \in A$ une lettre. Notons, comme précédemment, $m = s_1...s_k$ la factorisation en segments croissants maximaux de m, et $s_k = \underbrace{a_1...a_{i-1}}_{\alpha} \underbrace{a_i a_{i+1}...a_l}_{\beta}$ l'écriture du dernier segment, α et β vérifiant les hypothèses précisées plus haut (cf définition).

Cas 1 : $\alpha = s_k$. Alors x est ajouté à la fin de s_k, et on vérifie que, la forme du tableau ayant des lignes de longueur croissante, $(m \leftarrow a) = s_1...s_k a$ est toujours un mot de tableau, en se ramenant à la définition.

Cas 2 : $\alpha \neq s_k$. Alors, en notant a_h l'élément de m se trouvant juste au-dessus de a_i dans t. On a $a_h > a_i > a$. Ainsi, en remplaçant a_i par a dans la dernière ligne de t, on conserve la décroissance stricte suivant les colonnes. Il est également évident que la croissance large en ligne est conservée. Plus

généralement, on raisonne par récurrence finie évidente sur les insertions successives (au sens large), mais également suivant les colonnes (au sens strict).

Ainsi, on montre que $(m \leftarrow a)$ est un mot de tableau. □

Définition 14. On définit plus généralement l'insertion d'un mot $a = a_1...a_k$ dans un mot de tableau $m \in A*$ par :

$$(m \leftarrow a) = ((((m \leftarrow a_1) \leftarrow a_2)...) \leftarrow a_k)$$

C'est-à-dire l'insertion successive des lettres de a dans le mot m. De même, par une récurrence immédiate, l'insertion d'un mot dans un mot de tableau conserve le caractère de mot de tableau.

Deuxième partie

Relation de Knuth et monoïde plaxique

5 Relation de Knuth et insertion

5.1 Relation d'équivalence de Knuth

Définition 15. On définit la relation binaire suivante, que l'on appelera **relation de Knuth** (du nom de Donald Knuth - mathématicien américain, créateur de LaTeX) :

Soient $a, b \in A*$, ainsi que $x, y, z \in A$. On a alors : $\begin{cases} a(yzx)b \equiv a(yxz)b & si\ x < y \leq z \\ a(xzy)b \equiv a(zxy)b & si\ x \leq y < z \end{cases}$.

Moins formellement, deux lettres commutent si elles sont suivies ou précédées d'une lettre se trouvant entre elles pour l'ordre $\leq$. Deux mots satisfaisant à la relation $\equiv$ seront dits équivalents au sens de Knuth.

Plus généralement, on dira que deux mots sont équivalents/congrus au sens de Knuth s'il existe une suite finie de transformations de Knuth élémentaires permettant de passer de l'un à l'autre ;

Proposition 16. *La relation de Knuth est une relation d'équivalence sur $A*$ compatible avec la concaténation.*

Démonstration. On vérifie un à un les trois axiomes : transitivité, reflexivité et symétrie. □

5.2 Compatibilité avec l'insertion

Lemme 17. *La relation de Knuth est compatible avec l'insertion. On a, pour tout $a \in A$ et pour tout mot de tableau $m \in A*$:*

$$(m \leftarrow a) \equiv ma$$

Démonstration. On procède par récurrence sur le nombre de facteurs maximaux dans la décomposition du mot m. Notons $m = s_1 ... s_k$ et $s_k = \underbrace{a_1 ... a_{i-1}}_{\alpha} a_i \underbrace{a_{i+1} ... a_l}_{\beta}$. Comme précédemment, si $\alpha = s_k$, alors $(m \leftarrow a) = ma \equiv ma$. Supposons donc a_i bien défini. On effectue alors quelques transformations de Knuth élémentaires :

$$a_1 ... a_{i-1} a_i a_{i+1} ... a_l \mathbf{a} \equiv a_1 ... a_{i-1} a_i a_{i+1} ... a_{l-1} \mathbf{a} a_l \ \ car\ a < a_i \leq ... \leq a_{l-1} \leq a_l$$

$$a_1 ... a_{i-1} a_i a_{i+1} ... a_{l-1} \mathbf{a} a_l \equiv a_1 ... a_{i-1} a_i a_{i+1} ... a_{l-2} \mathbf{a} a_{l-1} a_l \ \ car\ a < a_i \leq ... \leq a_{l-2} \leq a_{l-1}$$

$$\vdots$$

$$a_1...a_{i-1}a_ia_{i+1}\mathbf{a}a_{i+2}...a_l \equiv a_1...a_{i-1}a_i\mathbf{a}a_{i+1}a_{i+2}...a_l \ \ car\ a < a_i \leq a_{i+1}$$

$$a_1...a_{i-1}a_i\mathbf{a}a_{i+1}a_{i+2}...a_l \equiv a_1...a_ia_{i-1}\mathbf{a}a_{i+1}...a_l \ \ car\ a_i > a \geq a_{i-1}$$

$$\vdots$$

$$a_1a_ia_2...a_{i-1}\mathbf{a}a_{i+1}...a_l \equiv a_ia_1...a_{i-1}\mathbf{a}a_{i+1}...a_l \ \ car\ a_1 \leq a_2 \leq a < a_i$$

Ce qui montre bien que, en reprenant les notations définies plus haut :

$$\alpha a_i\beta\mathbf{a} \equiv a_i\alpha\mathbf{a}\beta$$

On reconnait le motif définissant l'insertion, ce qui, par récurrence, permet de conclure sur la compatibilité de la relation de Knuth avec l'insertion d'une lettre dans un mot de tableau. □

Théorème 18. *Tout mot est équivalent, au sens de Knuth, à un mot de tableau.*

Démonstration. Ce résultat découle immédiatement du lemme précédent. En effet, le mot vide ε étant un mot de tableau, on a, en posant $m = a_1...a_n \in A*$ un mot, par une récurrence immédiate, $m \equiv (\varepsilon \leftarrow m)$. □

6 Sous-mots croissants et unicité

Définition 19. Soit $m \in A*$, $m = a_1...a_n$ un mot. On appelle sous-mot de m tout mot de la forme $a_{i_1}...a_{i_k}$ avec $1 \leq i_1 < ... < i_k \leq n$. On dira qu'un sous-mot est un sous-mot croissant s'il est croissant au sens large.

Exemple. Soit $m = 125632$. Alors 12, 122, 263 et 125632 sont des exemples de sous-mots de m.

Définition 20. Soit $m \in A*$ un mot et $k \in \mathbb{N}$. On définit $l_k(m)$ comme le maximum de la somme des longueurs de k sous-mots croissants disjoints de m. Plus formellement, en posant $S(m)$ l'ensemble des sous-mots croissants de m et, pour s un sous-mot de m, $I(s)$ l'ensemble des indices des lettres de s dans m et $L(s)$ la longueur de s, on a :

$$l_k(m) = max\ \left\{L(s_1) + ... + L(s_k) | (s_1, ..., s_k) \in S(m)^k\ et\ \forall (i,j) \in \{1, ..., k\}^2,\ i \neq j \Longrightarrow I(s_i) \cap I(s_j) = \emptyset\right\}$$

Exemple. On prend $m = 125632$. Il est assez facile de vérifier que le plus long sous-mot croissant de m est 1256. Ainsi, $l_1(m) = 4$. De même, on peut assez aisément se rendre compte que les deux sous-mots croissants disjoints ayant la longueur combinée la plus longue sont : 1256 et 23. Ainsi, on a $l_2(m) = 4 + 2 = 6$.

Lemme 21. *Soit $m \in A*$ un mot de tableau. On note $\lambda_1 \geq \lambda_2 \geq ...$ la longueur des lignes du tableau*

semi-standard associé à m. Alors on a :

$$\forall k \in \mathbb{N},\ l_k(m) = \lambda_1 + ... + \lambda_k = \sum_{i=1}^{k} \lambda_i$$

Démonstration. Il est clair que, le tableau t étant supposé semi-standard, un sous-mot croissant de m maximal "passe" nécessairement par chacune des colonnes de t, et ce exactement une fois. (croissance stricte de la suite des abscisses des lettres d'un sous-mot croissant). (*)

On démontre par récurrence sur j$\in \mathbb{N}$, le nombre de sous-mots croissants, le prédicat :

$P(j)$: "Quand j mots croissants disjoints de longueurs combinées maximales sont formés, il n'y a que $max(k_p - j, 0)$ cases libres dans la colonne p k_p désigne le nombre de lignes de longueur au moins p (ie la longueur de la colonne), et $p \in \{1, ..., \lambda_1\}$."

Initialisation : D'après (*), la longueur d'un sous-mot maximal croissant de m est au plus λ_k, puisque l'abscisse de sa dernière lettre inférieure ou égale à λ_k.

Hérédité : Soit $j \in \mathbb{N},\ k \geq 2$ tel que $P(j-1)$ soit vrai. Montrons que $P(j)$ l'est également.

On peut montrer simplement que, le tableau étant semi-standard, le j-ième sous-mot croissant maximal comprend exactement une case de t pour chaque abscisse $1, ..., \lambda_j$. Ainsi, pour tout $a \in \{1, ..., \lambda_j\}$, par hypothèse de récurrence, il n'y a plus que $max(k_a - j - 1, 0)$ cases libres à l'abscisse a.

Ainsi, $P(j)$ est vrai.

On obtient donc une majoration de $l_k(m)$ pour tout $k \in \mathbb{N}$. En effet, le raisonnement par récurrence précédent montre clairement que $\forall k \in \mathbb{N},\ l_k(m) \leq \lambda_1 + ... + \lambda_k$. Il suffit donc d'exhiber la liste des sous-mots, par hypothèse croissants et disjoints, formés par les lignes $1, ..., k$ de t, dont la somme des longueurs est égale à $\lambda_1 + ... + \lambda_k$, ce qui montre bien que $\forall k \in \mathbb{N},\ l_k(m) \geq \lambda_1 + ... + \lambda_k$, et termine la démonstration. □

Lemme 22. *Pour tout mot de tableau m, il existe $k_0 \in \mathbb{N}$ tel que $\forall k \geq k_0,\ l_k(m) = l_{k_0}(m)$.*

Démonstration. D'après le résultat précédent, la suite $(l_k(m))_{k\in\mathbb{N}}$ est une suite d'entiers croissante. De plus, il est clair que cette suite est majorée par la longueur du mot m. Par un résultat classique sur les suites, celle-ci est alors stationnaire, ce qui assure le résultat. □

Lemme 23. *Soient $(m, p) \in (A*)^2$ deux mots congrus au sens de Knuth, ie $m \equiv p$. Alors, pour tout $k \in \mathbb{N}$, on a $l_k(m) = l_k(p)$.*

Démonstration. Soit $k \in \mathbb{N}$. Il suffit de se ramener au cas des transformations de Knuth élémentaires, puis à généraliser ce résultat à des mots équivalents au sens de Knuth, car un nombre fini de transformations élémentaires permet de passer de l'un à l'autre. Considérons, sans perte de généralité (l'autre cas étant quasiment identique) la transformation élémentaire suivante :

$$m = \alpha\mathbf{yxz}\beta \equiv \alpha\mathbf{yzx}\beta = p$$

Considérons une sous-famille $(L_j)_{1\leq j\leq k}$ de sous-mots croissants disjoints de m et construisons, à partir de celle-ci, une famille de sous-mots croissants $(P_j)_{1\leq j\leq k}$ disjoints de p.

Dans la plupart des cas, la famille $(P_j)_{1\leq j\leq k}$ est la même que la famille $(L_j)_{1\leq j\leq k}$. Seul un seul cas doit être discuté : quand un des sous-mots, noté par exemple L_{j_0} est de la forme $\alpha\mathbf{xz}\beta$. En effet, la

commutation des deux lettres fait perdre à ce mot son caractère de sous-mot croissant. Il suffit alors d'opérer la disjonction de cas suivante :

(1) - Si aucun des autres mots (L_j) ne contient la lettre y, on remplace la lettre x dans L_{j_0} par la lettre y, ce qui permet de créer un sous-mot croissant de p tout en conservant le caractère disjoint de la famille construite.

(2) - S'il existe un mot, noté L_{j_1} de la famille (L_j) comportant la lettre y (ce mot est alors nécessairement unique), alors on remplace y par x dans L_{j_1} et x par y dans L_{j_0}, ce qui permet encore de conserver le caractère disjoint et croissant des sous-mots construits à partir de la famille considérée.

En généralisant par une récurrence évidente ce résultat à des mots équivalents au sens de Knuth, on achève la preuve. □

Lemme 24. *Soient u et v deux mots. On suppose u et v équivalents par la relation de Knuth, et on pose X la plus grande lettre apparaissant dans les deux mots. On appelle respectivement u' et v' les mots obtenus en supprimant la dernière occurence de X respectivement dans u et v.*

Alors $u' \equiv v'$.

Démonstration. Comme précédemment, il suffit de considérer le cas d'une transformation de Knuth élémentaire, comme

$$m = \alpha\mathbf{yxz}\beta \equiv \alpha\mathbf{yzx}\beta = p$$

Si la lettre effacée n'est pas x, y ou z, les deux mots obtenus en effaçant la lettre décrite plus haut sont encore congrus au sens de Knuth, par la définition même de cette relation.

En revanche, si la lettre effacée est l'une des trois lettres x, y ou z, alors, par définition de la relation d'équivalence de Knuth, la lettre effacée est forcément z, car la commutation des lettres x et z nécessite que $x < y \leq z$. Alors, en considérant les mots m et p privés de cette occurence de z, on vérifie que ces deux mots sont exactement les mêmes $(\alpha\mathbf{yx}\beta)$, et sont donc bien évidemment congrus au sens de Knuth, ce qui achève la démonstration. □

7 Théorème de la section et monoïde plaxique

Théorème 25. *(Théorème de la section) Tout mot est équivalent, par la relation d'équivalence de Knuth, à un seul et unique mot de tableau.*

Démonstration. L'existence est assurée par le Théorème 18. Il reste donc à montrer l'unicité.

Montrons que le tableau associé à un mot est unique, en le déterminant par la seule valeur de u. Démontrons ce résultat par récurrence sur n, la longueur du mot. On pose le prédicat $P(n)$: "Tout mot de longueur n est congru à un seul et unique mot de tableau".

Les cas $n = 0$ et $n = 1$ sont immédiats.

Soit $n \geq 2$ tel que $P(n-1)$ soit vrai. Montrons maintenant que $P(n)$ est vrai.

Soit u un mot de longeur n. On note u' le mot obtenu en retirant la copie la plus à droite de la plus grande lettre x figurant dans u, $t = (\emptyset \leftarrow u)$ et t' le mot obtenu en suivant le même procédé. On se donne également un mot de tableau a tel que $u \equiv a$, et a' le mot obtenu par le même procédé.

Ainsi, on a à la fois $t \equiv u$, $a' \equiv u'$ et $t' \equiv u'$ (conséquence du lemme précédent). Or, par hypothèse de récurrence (unicité), t' étant entièrement défini par la donnée de u', on a, les deux tableaux étant semi-standards, l'égalité $a' = t'$. La forme de a et t étant entièrement caractérisée par les sous-mots croissants (conséquence du lemme 23), x ne peut être placée qu'à un seul emplacement dans a' et t', ce qui permet de conclure : $a = t$.

Par le principe de récurrence, le mot de tableau est donc unique. □

Corollaire 26. *L'application τ, appelée redressement, qui associe à tout tableau de Young t l'unique tableau semi-standard t' vérifiant $\omega(t) \equiv \omega(t')$, est bien définie.*

Démonstration. La bonne définition de cette application est clairement assurée par le théorème précédent. □

Définition 27. On définit le **monoïde plaxique**, noté M comme l'ensemble quotient de $A*$ par la relation d'équivalence de Knuth. Plus précisément, chaque classe d'équivalence sera représentée par le seul et unique mot de tableau qu'elle contient.

Troisième partie

Correspondance de Robinson-Schensted-Knuth

8 Bi-mots sur un couple d'alphabets finis

Définition 28. Soient $\mathbb{A}$ et $\mathbb{B}$ deux alphabets finis totalement ordonnés. On appelle bi-mot un élément de $(\mathbb{A}\times\mathbb{B})*$, et l'on note un tel élément $u=\begin{pmatrix} a_1 & a_2 & \cdots & \cdots & a_n \\ b_1 & b_2 & \cdots & \cdots & b_n \end{pmatrix}$.

Les lettres (a_i) seront appelées lettres hautes de u et les lettres (b_i), les lettres basses.

On définit sur l'ensemble des bi-lettres un ordre lexicographique standard :
$\begin{pmatrix} a_1 \\ b_1 \end{pmatrix} \preceq \begin{pmatrix} a_2 \\ b_2 \end{pmatrix} \Longleftrightarrow (a_1 < a_2)\ ou\ (a_1 = a_2\ et\ b_1 \leq b_2)$ - qui définit bien entendu un ordre total sur l'ensemble des bi-lettres. Plus généralement, on dira qu'un bi-mot est lexicographique quand les bi-lettres qui le composent sont rangées dans l'ordre lexicographique, et on notera $L_n(\mathbb{A},\mathbb{B})$ l'ensemble des bi-mots lexicographiques à n bi-lettres.

Exemple. On définit les alphabets finis $\{1,...,n\}$ et $\{A,B,...,Z\}$. On peut se donner, à titre d'exemple, les bi-mots suivants :
$\begin{pmatrix} 1 & 4 & 5 & 2 & 3 \\ R & T & A & C & V \end{pmatrix}, \begin{pmatrix} 3 & 1 & n \\ A & C & V \end{pmatrix}$ et $\begin{pmatrix} 1 \\ C \end{pmatrix}$

Définition 29. On définit $\Xi(\mathbb{A},\mathbb{B})$ comme l'ensemble des couples de tableaux de Young semi-standards appartenant à $T(\mathbb{A})\times T(\mathbb{B})$ de même forme.

9 Correspondance de Robinson-Schensted-Knuth

Définition. On définit de la manière suivante la correspondance RSK (Robinson-Schensted-Knuth) :

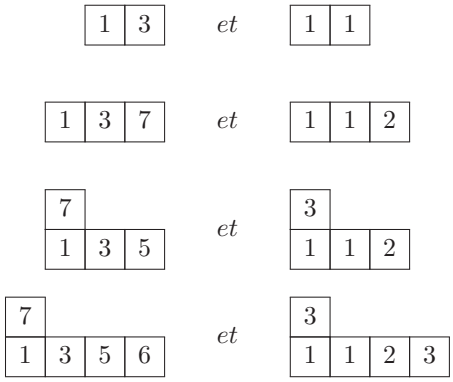
Soit $A=\begin{pmatrix} a_1 & a_2 & a_3 & \cdots & \cdots & a_{n-1} & a_n \\ b_1 & b_2 & b_3 & \cdots & \cdots & b_{n-1} & b_n \end{pmatrix}$ un bi-mot lexicographique. On associe à A un couple (P,Q) de tableaux en considérant les cas suivants :

(i) - Si $A=\epsilon$ (A est le mot vide), alors P et Q sont les tableaux vides : $P=\emptyset$ et $Q=\emptyset$.

(ii) - Sinon, on écrit $A=A'\begin{pmatrix} a_n \\ b_n \end{pmatrix}$, où $\begin{pmatrix} a_n \\ b_n \end{pmatrix}$ est la dernière lettre du bi-mot A, et, récursivement, en notant P' et Q' les tableaux associés à A', on pose $P=(P'\leftarrow b_n)$, et Q est obtenu en ajoutant à Q' la lettre a_n dans une nouvelle case, correspondant à la case nouvellement créée dans P lors de l'insertion de b_n (qui n'est pas nécessairement la case contenant b_n).

Exemple. Appliquons la correspondance RSK au bi-mot lexicographique $\begin{pmatrix} 1 & 1 & 2 & 3 & 3 \\ 1 & 3 & 7 & 5 & 6 \end{pmatrix}$. On obtient successivement les couples de tableaux suivants :

$\boxed{1}$ *et* $\boxed{1}$



Proposition. *La correspondance RSK* (ϖ) *définit une application de l'ensemble des bi-mots lexicographiques de* $(\mathbb{A} \times \mathbb{B})*$ *vers* $\Xi(\mathbb{A}, \mathbb{B})$.

Démonstration. Démontrons tout d'abord que la correspondance RSK donne, pour tout bi-mot lexicographique un couple de tableaux semi-standards, c'est-à-dire montrons la bonne définition de la correspondance RSK. Notons (P, Q) le couple de tableaux obtenus par l'application de la correspondance RSK sur un bi-mot $\begin{pmatrix} a_1 & a_2 & \cdots & a_n \\ b_1 & b_2 & \cdots & b_n \end{pmatrix}$. Il est clair que, étant issu de l'insertion répétée de lettres dans un tableau semi-standard, le tableau P est semi-standard.

Montrons qu'alors le tableau Q est également semi-standard.

Premièrement, il est évident, au vu de la construction du tableau Q, que celui-ci est de la même forme que P, et donc que sa forme est celle d'un tableau semi-standard. Montrons qu'il vérifie la condition de décroissance stricte en colonnes et de croissance large en lignes. Démontrons ce résultat par récurrence finie. Notons, pour $k \in \{0, ..., n\}$, P_k et Q_k les tableaux obtenus par l'application respective de la correspondance de Robinson-Schensted-Knuth sur le bi-mot composé des k premières bi-lettres de A. Posons alors, pour $k \in \{0, ..., n\}$,$P(k)$:"Q_k est un tableau semi-standard".

L'initialisation est évidente : $Q_0 = \emptyset$ est bien un tableau semi-standard.

Soit $k \in \{1, ..., n\}$ tel que $P(k-1)$ soit vrai. Montrons alors que $P(k)$ est vérifié.

$P_k = (P_{k-1} \leftarrow b_k)$ est un tableau semi-standard, et Q_{k-1} est également un tableau semi-standard. On obtient Q_k en insérant a_k dans une nouvelle case de Q_{k-1} située à l'emplacement où se trouve la case créée par l'insertion de b_k dans P_{k-1}. Remarquons que, par définition de l'opération d'insertion d'une lettre dans un tableau, la case nouvellement créée lors de l'insertion se situe forcément en bout d'une ligne ou tout en haut (au-dessus du tableau). Le caractère lexicographique du bi-mot nous permet de distinguer les deux cas suivants :

Cas 1 : $a_{k-1} < a_k$. Plus précisément, $a_0 \leq a_1 \leq ... \leq a_{k-1} < a_k$.

Ainsi, l'insertion de la lettre a_k en haut ou au bout d'une ligne du tableau Q_{k-1}conservera forcément la décroissance stricte en colonnes et la croissance au sens large en lignes. Ainsi, le tableau Q_k obtenu ayant la forme d'un tableau semi-standard (car sa forme est la même que celle de P_k, qui est un tableau de Young semi-standard) et vérifiant les conditions de croissance, il est encore un tableau semi-standard, ce qui termine ce cas.

Cas 2 : $a_{k-1} = a_k$ et alors $b_{k-1} \leq b_k$.

Alors la forme de Q_k est celle d'un tableau semi-standard, la croissance en lignes est conservée car a_k est maximal, le mot A étant ordonné lexicographiquement. Vérifions la décroissance stricte en colonnes. Supposons, par l'absurde, que le a_k ajouté soit situé au-dessus d'une case contenant $a_p = a_k$. Alors, le tableau est de type :

...	...	...	...	...			
...	...	...	...	a_k	...	...	
...	...	...	...	a_p	...	...	...

Or, b_p est inférieur ou égal à b_k car $a_p = ... = a_k$ (le bi-mot étant lexicographique). Ainsi, en insérant b_k dans le tableau Q_{k-1}, et d'après le lemme (12), on aura $\Pi(Q_{p-1}, b_p)$ de longueur supérieure à $\Pi(Q_p, b_{p+1})$, etc..., de longueur supérieure à $\Pi(Q_{k-1}, b_k)$ car $b_p \leq ... \leq b_k$. Par transitivité, comme $b_p \leq b_{p+1} \leq ... \leq b_k$, la case contenant a_k ne pourra pas se situer au-dessus de celle contenant a_p. C'est absurde, donc la décroissance stricte en colonnes est vérifiée. Ainsi, Q_k est semi-standard.

Ainsi, $P(k)$ est vrai. Par le principe de récurrence, on a en particulier la véracité de $P(n)$, ce qui permet de conclure quant à la nature des tableaux P et Q. Ainsi, la correspondance RSK est bien définie. □

Lemme 30. *Soit $(P \leftarrow a)$ un tableau semi-standard ($a \in \mathbb{A}$). Alors il existe une application $\Sigma : ((P \leftarrow a), p_k) \mapsto p_{k-1}$ où p_k est le k-ième élément de la perturbation associée à l'insertion de a dans P, avec $k > 0$.*

Démonstration. On considère un tableau de Young semi-standard $(P \leftarrow a)$, avec a une lettre et P un tableau semi-standard. On se donne la connaissance de p_k. Retrouvons p_{k-1}. Par définition de l'insertion, lors de l'insertion de p_{k-1} dans la ligne L_{k-1} de P, on a$p_k = min\ \{l \in L_{k-1}(P) | l > p_{k-1}\}$. Ainsi, on a $p_{k-1} = max\ \{l \in L_{k-1}(P \leftarrow a) | l < p_k\}$, en prenant la case la plus à droite (maximum qui est bien défini), ce qui termine la démonstration. □

Lemme 31. *Soit $(P \leftarrow a)$ un tableau semi-standard ($a \in \mathbb{A}$). Alors il existe une application $\Sigma : ((P \leftarrow a), p_n) \mapsto a$ où p_n est le dernier élément de la perturbation associée à l'insertion de a dans P.*

Démonstration. La bonne définition est assurée par récurrence immédiate en utilisant le lemme précédent. □

Théorème 32. *La correspondance de Robinson-Schensted-Knuth définit une bijection de l'ensemble des bi-mots lexicographiques de $(\mathbb{A} \times \mathbb{B})*$ dans $\Xi(\mathbb{A}, \mathbb{B})$.*

Démonstration. Mettons en évidence la bijection réciproque du procédé de Robinson-Schensted-Knuth.

Explicitons tout d'abord, sans formaliser, le procédé utilisé pour, à partir d'un couple (P, Q) de tableaux semi-standards de même forme, reconstruire le bi-mot original.

L'idée générale du procédé est d'inverser l'étape d'insertion de b_k dans le mot P_{k-1}. Pour ce faire, on considère l'élément b_p situé dans P_k à l'emplacement où se trouve a_k dans Q_k. On sait que a_k est la copie la plus à droite de la valeur maximale présente dans Q_k. Pour retrover b_k, on sait que b_p a été déplacé par le plus grand b_{p_0} strictement inférieur à b_p situé dans la ligne sous la case contenant b_p. De même, on applique à nouveau ce procédé à b_{p_0}, en construisant une suite finie (b_{p_i}) d'éléments, jusqu'à arriver à la dernière ligne du tableau Q_k. Le dernier élément de cette suite sera alors b_k. Formalisons quelque peu ce résultat.

Montrons qu'à partir des tableaux P_k et Q_k, il est possible de retrouver a_k et b_k. Il suffit, pour trouver a_k, de rechercher dans Q_k la copie la plus à droite de la plus grande lettre (pour l'ordre dans l'alphabet $\mathbb{A}$). Ensuite, on considère la lettre b_p située dans P_k au même emplacement que a_k. Comme on connaît la position de la dernière lettre de la perturbation associée à l'insertion de b_k dans P_k, le lemme (31) nous permet de retrouver b_k. Ainsi, en supprimant a_k et b_k respectivement des tableaux Q_k et P_k, on peut conclure par récurrence finie.

La correspondance de Robinson-Schensted-Knuth admet donc une bijection réciproque et est donc bijective. Ainsi, elle constitue une bijection canonique entre l'ensemble des bi-mots lexicographiques et celui des couples de tableaux semi-standards de même forme. □

Proposition 33. *Le résultat précédent permet d'arriver à la formule suivante :*

$$Card(\Xi_n(\mathbb{A},\mathbb{B})) = \begin{pmatrix} Card(\mathbb{A}) \times Card(\mathbb{B}) + n - 1 \\ n \end{pmatrix}$$

où $\Xi_n(\mathbb{A},\mathbb{B})$ est l'ensemble des couples de tableaux de Young semi-standards à n cases à valeurs respectivement dans $\mathbb{A}$ et $\mathbb{B}$.

Démonstration. La bijection établie précédemment entre $\Xi_n(\mathbb{A},\mathbb{B})$ et $L_n(\mathbb{A},\mathbb{B})$ nous permet de dénombrer facilement ce premier ensemble en dénombrant le nombre de suites finies croissantes pour l'ordre lexicographique d'éléments de l'alphabet fini $\mathbb{A} \times \mathbb{B}$. Dénombrons ces suites.

$\mathbb{A}$ et $\mathbb{B}$ sont deux ensembles finis, donc $Card(\mathbb{A} \times \mathbb{B}) = Card(\mathbb{A}) \times Card(\mathbb{B}) = N$. Notons $c_1, ..., c_N$ les éléments de $\mathbb{A} \times \mathbb{B}$ ordonnés par l'ordre lexicographique croissant. On met alors $\mathbb{A} \times \mathbb{B}$ en bijection naturelle avec $\{1, ..., N\}$. Il nous suffit donc de dénombrer les suites croissantes au sens large de $\{1, ..., N\}$. Notons donc, pour a et b deux entiers naturels quelconques $C_n(a)$ l'ensemble des suites croissantes au sens large de $\{1, ..., a\}$ à n éléments, ainsi que $SC_n(b)$ l'ensemble des suites strictement croissantes de $\{1, ..., b\}$ à n éléments. On pose alors les applications suivantes, à n fixé :

$$\phi : \begin{pmatrix} C_n(N) \to SC_n(N+n-1) \\ (e_1, ..., e_n) \longmapsto (e_1, e_2 + 1, ..., e_n + n - 1) \end{pmatrix}$$

$$\psi : \begin{pmatrix} SC_n(N+n-1) \to C_n(N) \\ (e_1, ..., e_n) \longmapsto (e_1, e_2 - 1, ..., e_n - n + 1) \end{pmatrix}$$

On vérifie que ϕ et ψ sont bien définies, et que $\phi \circ \psi = id_{SC_n(N+n-1)}$ ainsi que $\psi \circ \phi = id_{C_N(N)}$. Ainsi, ϕ et ψ sont bijections réciproques. On a donc $Card(C_n(N)) = Card(SC_n(N+n-1))$. Or, on met facilement en bijection l'ensemble des suites strictement croissantes à n éléments de $\{1, ..., N+n-1\}$ avec l'ensemble des parties à n élément de ce même ensemble. Il vient donc :

$$Card(C_n(N)) = \begin{pmatrix} N+n-1 \\ n \end{pmatrix}$$

D'où, finalement :

$$Card(\Xi_n(\mathbb{A},\mathbb{B})) = \left(\begin{array}{c} Card(\mathbb{A}) \times Card(\mathbb{B}) + n - 1 \\ n \end{array} \right)$$

□

Quatrième partie

Application à l'étude des permutations

Intéressons-nous maintenant à un cas particulier, historiquement à l'origine du théorème précédent, qui nous permettra de caractériser la plus longue sous-séquence croissante d'une permutation et de proposer une démonstration plutôt originale du théorème d'Erdös-Szekeres.

10 Théorème de Robinson-Schensted et identité de Frobenius

10.1 Théorème de Robinson-Schensted

Théorème 34. *(Théorème de Robinson-Schensted)*

*Soit $\sigma \in S_n$ une permutation de $\{1, ..., n\}$. On reprend la notation présentée précédemment pour une permutation : $\sigma : \begin{pmatrix} 1 & 2 & \cdots & \cdots & n \\ \sigma(1) & \sigma(2) & \cdots & \cdots & \sigma(n) \end{pmatrix}$. Alors il existe une bijection entre l'ensemble des permutations de $\{1, ..., n\}$ et l'ensemble des couples de tableaux **standards** de même forme. Cette bijection est donnée par l'application de la correspondance RSK.*

Démonstration. Il suffit de démontrer que la correspondance RSK, appliquée à un tel bi-mot, donne un tableau standard et non simplement semi-standard. En appliquant cette correspondance à un bi-mot représentant une permutation, on obtient un couple de tableaux semi-standards de même forme. Or, ce bi-mot représentant une permutation, les lettres hautes et basses de celui-ci forment deux suites d'éléments deux à deux distincts. Ainsi, la croissance large en lignes se change naturellement en croissance au sens strict. Le tableau considéré n'est donc pas seulement semi-standard. Il est standard. □

10.2 Identité de Frobenius

Théorème 35. *(Identité de Frobenius ou formule de la somme des carrés)*

La bijection donnée par le théorème de Robinson-Schensted mène à l'identité suivante :

$$\forall n \in \mathbb{N},\ n! = \sum_{\lambda \in P_n} (f^\lambda)^2$$

où P_n et l'ensemble des partitions de n (vues commes des partages à n cases) et f^λ le nombre de tableaux de Young standards de forme λ.

Démonstration. Soit $n \in \mathbb{N}$. Il est bien connu que le cardinal du groupe symétrique à n éléments est $n!$.

Il suffit donc de dénombrer le nombre de couples de tableaux de Young semi-standards de même forme. Ecrivons cet ensemble, que l'on notera Δ_n, sous la forme d'une réunion disjointe, ce qui nous permettra de conclure. On a, en décrivant cet ensemble par forme des couples :

$$\Delta_n = \bigcup_{\lambda \in P_n} \{(Y_1, Y_2) \in T_s^2 | \lambda(Y_1) = \lambda \text{ et } \lambda(Y_2) = \lambda\}$$

Cette réunion étant disjointe, on a :

$$Card(\Delta_n) = \sum_{\lambda \in P_n} Card(\{(Y_1, Y_2) \in T_s^2 | \lambda(Y_1) = \lambda \; et \; \lambda(Y_2) = \lambda\})$$

Or, on a :

$$\{(Y_1, Y_2) \in T_s^2 | \lambda(Y_1) = \lambda \; et \; \lambda(Y_2) = \lambda\} = (T_s^\lambda)^2$$

En notant T_S^λ l'ensemble des tableaux standards de forme λ. Ainsi, on a bien, en utilisant la bijection donnée par le théorème de Robinson-Schensted :

$$\forall n \in \mathbb{N}, \; n! = \sum_{\lambda \in P_n} (f^\lambda)^2$$

□

11 Sous-suites monotones d'une permutation

11.1 Définitions

Définition 36. Soit $\sigma \in S_n$ une permutation. On note $\sigma : \begin{pmatrix} 1 & 2 & \cdots & \cdots & n \\ a_1 & a_2 & \cdots & \cdots & a_n \end{pmatrix}$. On définit le **retourné** de σ, que l'on note σ^r par $\sigma^r : \begin{pmatrix} 1 & 2 & \cdots & \cdots & n \\ a_n & a_{n-1} & \cdots & \cdots & a_1 \end{pmatrix}$. Le retourné d'une permutation est encore une permutation.

Exemple. On considère la permutation $\sigma : \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 3 & 4 & 1 & 2 & 5 \end{pmatrix}$. On aura alors $\sigma^r : \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 5 & 2 & 1 & 4 & 3 \end{pmatrix}$, qui est toujours une permutation.

Définition 37. Soit T un tableau de Young de partage d, à valeurs dans un alphabet $\mathbb{A}$. On définit alors le **transposé** de T, noté ${}^tT$, par :

$${}^tT := \begin{pmatrix} d^c \to \mathbb{A} \\ (i,j) \longmapsto T(j,i) \end{pmatrix}$$

Exemple. On considère le tableau suivant :

10		
7	6	
1	3	5

Son transposé est alors :

5		
3	6	
1	7	10

Proposition 38. *Le transposé d'un tableau de Young standard est toujours un tableau standard.*

Démonstration. Le support de ${}^tT$ estd^c, qui est toujours un partage. De plus, la croissance stricte de T selon les lignes assure la croissance stricte de ${}^tT$ selon les colonnes, et la croissance stricte selon les colonnes de T assure la croissance stricte en lignes de ${}^tT$. Enfin, ${}^tT$ est à valeurs dans $\{1, ..., n\}$, tout comme T. Donc ${}^tT$ est un tableau de Young standard. □

11.2 Quelques lemmes

Lemme 39. *Le retournement commute avec l'équivalence de Knuth pour les mots standards. Soient u et v deux mots. Alors $u \equiv v \iff u^r \equiv v^r$.*

Démonstration. Il suffit de considérer les transformations de Knuth élémentaires pouvant avoir lieu dans le cas d'un mot standard, c'est-à-dire celles impliquant des inégalités strictes (les lettres d'un mot standard étant distinctes deux à deux). Montrons que $u \equiv v \iff u^r \equiv v^r$.

Supposons $u \equiv v$. Considérons le cas où v diffère de u d'une seule transformation de Knuth élémentaire. Alors il existe α, β deux mots et x, y, z trois lettres telles que :

$$u = \alpha xyz\beta$$

et telles que $v = \alpha xzy\beta$ (si $y < x < z$) ou $v = \alpha yxz\beta$ (si $x < z < y$). On a également $u^r = \beta^r zyx\alpha^r$.

Cas 1 : $y < x < z$. Alors $v = \alpha xzy\beta$ donc $v^r = \beta^r yzx\alpha^r$. Or, $y < x < z$ donc $yzx \equiv zyx$, d'où $v^r \equiv u^r$.

Cas 2 : $x < z < y$. Alors $v = \alpha yxz\beta$ donc $v^r = \beta^r zxy\alpha^r$. Or, $x < z < y$ donc $zxy \equiv zyx$, d'où $v^r \equiv u^r$.

L'application $u \longmapsto u^r$ étant une involution, le raisonnement est totalement symétrique et on peut donc conclure sur l'équivalence en remarquant que les lettres de u^r sont toujours deux à deux distinctes si u est un mot standard. □

Lemme 40. *Soit $\sigma : \begin{pmatrix} 1 & 2 & \cdots & \cdots & n \\ a_1 & a_2 & \cdots & \cdots & a_n \end{pmatrix}$ une permutation. On note $(P, Q) = \varpi(\sigma)$, et $(P', Q') = \varpi(\sigma^r)$; on a alors $P' = {}^t P$.*

Démonstration. Notons $\omega(P) = a_1...a_n$ avec P le tableau associé à σ par la correspondance de Robinson-Schensted. Montrons que le redressé de $\omega(P)^r$ est égal à $\omega({}^tP)$.

Construisons par récurrence le redressé de $\omega(P)^r = m$. On note $m = s_1...s_k$ la décomposition en segments **décroissants** maximaux de m. On a $\forall i,\ s_i = l_i^r$, où l_i est la i-ième ligne du tableau P.

On note, pour $i \in \{1, ..., k\}$, T_i le tableau associé à s_i (qui est une colonne et donc un tableau standard, s_i étant un mot strictement décroissant).

Travaillons par double récurrence finie, entre et à l'intérieur des s_a.

Premier prédicat : Au rang i, le tableau associé au mot de tableau $(((\epsilon \leftarrow s_1) \leftarrow s_2)....\leftarrow s_i)$ est de la forme :

T_1	T_2	...	T_i

Second prédicat : on considère la j-ième lettre de s_i. Notons $T''_{i,j}$ le tableau lié à l'insertion dans le mot vide des j premières lettres de s_i. A l'insertion de cette lettre, le tableau est de la forme :

T_1	T_2	...	T_{i-1}	$T'_{i,j}$

L'initialisation du premier prédicat est immédiate. Montrons son hérédité, en le supposant vrai au rang $i-1$ ($i \geq 1$). Pour montrer celle-ci, montrons la véracité du second prédicat pour tout j.

L'initialisation du second prédicat est assurée par hypothèse de récurrence sur le premier prédicat.

Supposons, le second prédicat vrai au rang $j-1$, pour $j \geq 1$. Montrons sa véracité au rang j. On insère $s_{i,j}$ la j-ième lettre de s_i dans le tableau. Le tableau, avant l'insertion, est de la forme :

$s_{1,1}$			
$s_{1,2}$	$s_{2,1}$		
...	...	...	
...	...	...	$s_{i,1}$
...	...	...	...
s_{1,l_1}	...	...	$s_{i,j-1}$

Comme P est un tableau de Young standard, on a : $s_{i,j} > s_{i,1}$ (dû à la croissance stricte de P selon ses lignes) et $s_{i,1} > s_{i-1,l_{i-1}}$ (dû à la stricte décroissance de P selon ses colonnes). D'où $s_{i-1,l_{i-1}} < s_{i,j} < s_{i,j-1}$ car s_i est un mot décroissant. Pour tout $k \in \{1, ..., l_{i-1}-1\}$, l'inégalité $s_{i-1,l_{i-1}-k} < s_{i,l_{i-1}-k} < s_{i,j-k}$ est vérifiée. Donc, par définition de l'insertion (on ne détaillera pas ici la triple récurrence nécessaire), le deuxième prédicat est héréditaire, d'où, en $j = l_i$), on montre l'hérédité du premier prédicat. Ainsi, par le principe de récurrence, au rang k, on a un tableau de la forme :

T_1	T_2	...	T_k

Et on vérifie que ce tableau est exactement le transposé de P. □

11.3 Théorème principal

Théorème 41. *Soit $\sigma \in S_n$ une permutation. Notons $\sigma : \begin{pmatrix} 1 & 2 & \cdots & \cdots & n \\ a_1 & a_2 & \cdots & \cdots & a_n \end{pmatrix}$ ainsi que $(P, Q) = \varpi(\sigma)$. Alors la longueur de la plus longue sous-séquence croissante de σ est égale à la longueur de la dernière ligne de P. De même, la longueur de la plus longue sous-séquence décroissante de σ est égale à la longueur de la première colonne de P.*

Démonstration. On utilise en particulier les lemmes 21 et 23. On considère le mot de tableau m_1 associé au tableau P, ainsi que le mot $m = a_1 a_2 ... a_n$, naturellement associé à la permutation σ. Par

définition de l'algorithme de Robinson-Schensted, ces deux mots sont congrus au sens de Knuth, m_1 résultant du redressement de m. Ainsi, par application du lemme 23, on $l_1(m) = l_1(m_1)$. m_1 étant un mot de tableau, le lemme 21 assure que $l_1(m_1) = \lambda_1$. Ainsi, la longueur de la plus longue sous-séquence croissante de σ est égale à la longueur de la dernière ligne de P.

De même, remarquons que la plus longue sous-séquence décroissante de σ est exactement la plus longue sous-séquence croissante de σ^r. Considérons alors le tableau de Young P' résultant de l'application de l'algorithme de Robinson-Schensted à la permutation σ^r. Alors, la plus longueur de la plus longue sous-séquence croissante de σ^r est égale à la longueur de la dernière ligne de P'. Or, d'après le lemme précédent, on a $P' =^t P$. Ainsi, la longueur de la plus longue sous-séquence décroissante de σ est égale à la longueur de la première colonne de P. □

11.4 Corollaire : Théorème d'Erdös-Szekeres

Théorème 42. *(Théorème d'Erdös-Szekeres)*

Soient $(p, q) \in (\mathbb{N}*)^2$, et $\sigma \in S_{pq+1}$. Alors σ admet une sous-séquence croissante de longueur supérieure ou égale à $p + 1$ ou une sous-séquence croissante de longueur supérieure ou égale à $q + 1$.

Démonstration. Considérons le tableau de Young P résultant de l'application de l'algorithme de Robinson-Schensted à la permutation σ. Supposons que σ n'admette pas de sous-séquence croissante de longueur au moins $p + 1$. Alors, par la contraposée du théorème (42), P est de largeur au plus p et contient $pq + 1$ éléments. En supposant, par l'absurde, que P soit de hauteur $h < q + 1$, alors P contient au plus $hp < pq + 1$ éléments, ce qui est absurde. Ainsi, P est de hauteur supérieure ou égale à $q + 1$. Donc sa première colonne est de longueur supérieure ou égale à $q + 1$, ce qui garantit, d'après le théorème (42), l'existence d'une sous-séquence décroissante de longuer au moins $q + 1$.

Ainsi, σ admet une sous-séquence croissante de longueur au moins $p + 1$, ou une sous-séquence décroissante de longueur au moins $q + 1$. □

Exemple. Considérons, à titre d'exemple, la permutation suivante :

$$\sigma : \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ 3 & 5 & 2 & 9 & 6 & 8 & 7 & 4 & 1 \end{pmatrix}$$

C'est une permutation de $\{1, .., 9\}$, et $9 = 2 \times 4 + 1$. Ainsi, d'après le théorème d'Erdös-Szekeres, elle admet une séquence croissante de longueur au moins 5, ou une séquence décroissante de longueur au moins 3 (ou l'inverse). C'est en effet le cas : 98741 est une sous-séquence décroissante de longueur 5 de σ.

Cinquième partie

Annexes

11.5 Algorithme de redressement (Python)

On propose l'algorithme présenté plus bas, réalisant la fonction d'insertion en Python de façon récursive et naturelle. Il permet d'obtenir, à partir d'un mot quelconque, l'unique mot de tableau auquel il est congru au sens de la relation d'équivalence de Knuth.

11.6 Algorithme de Robinson-Schensted (CamL)

On propose l'algorithme présenté plus bas, donnant le mot de tableau associé au tableau résultant de l'application de l'algorithme de Robinson-Schensted à une permutation. Il implémente également la fonction d'insertion de façon récursive, en suivant la définition de celle-ci.

11.7 Etude de la complexité de l'algorithme de Robinson-Schensted

Considérons la complexité au pire de cet algorithme. L'algorithme effectue au pire autant de tests qu'il n'y de lettres dans le mot, pour chaque étape d'insertion. Ainsi, en notant n la longueur du mot à redresser, on obtient une complexité au pire en $\sum_{k=1}^{n} k = O(n^2)$. En pratique, on vérifie que cette complexité est réalisée pour une permutation strictement décroissante. Ainsi, il est possible de déterminer en un temps quadratique la longueur de la plus longue sous-séquence croissante/décroissante d'une permutation.

Comparons cette complexité à celle de l'algorithme naïf permettant de déterminer la plus longue sous-séquence croissante/décroissante d'une permutation. Cet algorithme doit tester chaque sous-suite de la permutation, soit 2^n possibilités et, pour chacune d'entre elles, tester si elle est bien croissante/décroissante et vérifier sa longueur. Sa complexité est donc catastrophique, de type exponentielle. Ainsi, l'algorithme de Robinson-Schensted apporte une amélioration non négligeable à la version naïve.

11.8 Bibliographie

Références

[1] I. Kortchemski, *Bonnes suites, sous-suites croissantes et tableaux de Young*, Juin 2006

[2] A. Lascoux & M.P. Schützenberger, *Le monoïde plaxique* (issu de *Non-commutative structures in Algebra and Geometric combinatorics*), paru dans *La ricerca scientifica*, n°109, 1981

[3] D. Knuth, *Permutations, matrices and generalized Young tableaux*, paru dans *Pacific Journal of Mathematics*, Vol. 34, n°3, 1970

[4] F. Bergeron, *Combinatoire algébrique*, Cours de l'Université de Montréal, 2001

[5] C. Schensted, *Longest decreasing and increasing subsequences*, paru dans *Canadian Journal of Mathematics*, n°13, 1961

[6] F. Wlazinski, *Cours de L2 - Structures algébriques usuelles*, Cours de l'Université de Picardie, 2005

[7] L. Pirutka & N. de Saxcé, *Les représentations du groupe symétrique*, Cours de l'ENS

[8] B. Sagan, *The Symmetric Group*, New York, 2001

[9] G. C. Panova, *Combinatorial applications of symmetric function theory to certain classes of permutations and truncated tableaux*, Harvard University, 2011

```python
def get_lastWord(mot):
    #Fonction récupérant le dernier sous-mot croissant d'un mot quelconque
    #Renvoie le dernier sous-mot croissant de ce mot, ainsi que la séquence précédent
    sous_mot = []
    mot_avant = []
    for i in enumerate(mot):
        if sous_mot != [] and i[1] >= sous_mot[len(sous_mot)-1]:
            sous_mot.append(i[1])
        else:
            sous_mot = []
            if i[0]>0:
                mot_avant = mot[:i[0]]
            sous_mot.append(i[1])
    return sous_mot, mot_avant

def insertion(mot_init, x):
    #Fonction permettant d'insérer une lettre dans un mot de tableau.
    mot = list(mot_init)
    if mot == []:
        return [x]
    lastWord, head = get_lastWord(mot)
    if x >= lastWord[len(lastWord)-1]:
        mot.append(x)
        return mot
    i = 0
    while lastWord[i]<=x:
        i+=1
    x_prime = lastWord[i]
    lastWord[i] = x
    head = insertion(head,x_prime)
    return head+lastWord

def redressement(mot):
    #Utilise récursivement la fonction d'insertion pour redresser un mot.
    T = []
    for e in mot:
        T = insertion(T,e)
    return T

def dispYoung(mot2):
    liste = []
    mot = list(mot2)
    while mot!=[]:
        lastWord,head = get_lastWord(mot)
        mot = head
        liste.append(lastWord)
    liste.reverse()
    for i in liste:
        print i
```

```
(*** Concatène deux listes ***)

let rec concatene l1 l2 =
    match l1,l2 with
    |[],l2 -> l2
    |a::w,l2 ->a::(concatene w l2) ;;

(*** Retourne une liste ***)
let miroir l1 =

    let rec miroir_aux l1 l2 =
        match l1,l2 with
            |a,[] -> a
            |l1,a::l2 -> miroir_aux (a::l1) l2
    in miroir_aux [] l1;;

(*** Permet d'accéder au dernier élément d'une liste ***)
let rec dernière_lettre = function
    |[] -> 1
    |[a] -> a
    |a::q -> dernière_lettre q ;;

(*** Permet de dénombrer le nombre d'occurrences d'une lettre dans un mot ***)

let rec nombres_d_occurrences lettre mot =

    match lettre,mot with
        |a,[] -> 1
        |x,a::mot -> (nombres_d_occurrences x mot) + if a=x then 1 else 0 ;;

(*** Permet de trouver Teta(k) ***)

let dernière_sous_suite w =

    let rec aux w dernière = match w,dernière with
        |[],dernière-> [],miroir(dernière)
        |a::q,[]->aux q [a]
        |a::q,a2::q2-> if a2>=a then aux q (a::dernière) else miroir(w),dernière
    in aux (miroir w) [];;


(*** Permet de récupérer x' et de retourner Theta(k)' ***)

let insère_bis dernière x =

    let rec aux dernière x sac =
    match dernière,x with
    |[],x->x,[]
    |(a::q),x-> if a<=x then aux q x (a::sac) else a,(concatene (miroir sac) (x::q))
    in aux dernière x [];;


(*** On définit l'insertion comme elle est définie récursivement ***)

let rec insère w x = let (mot_sans,dernière) = dernière_sous_suite w in
                if x >= dernière_lettre(dernière) then concatene w [x] else
                let y,dernière_seconde = insère_bis dernière x in
                concatene (insère mot_sans y) (dernière_seconde) ;;
```

```
(*** Algorithme de Robinson-Schensted ***)

let robinson w =

    let rec aux tableau = function
        |[]->tableau
        |a::q->aux (insère tableau a) q

    in aux [] w;;

(*** On peut tester Robinson ***)

robinson [6;4;1;21;65;4;32;54;4;65;8;9;7;8];;
```