THE CHINESE MONOID

Julien Cassaigne*
Marc Espie†
Florent Hivert‡
Daniel Krob§
Jean-Christophe Novelli¶

November 14, 1995

Résumé

Cet article présente une étude combinatoire du monoïde Chinois, un monoïde ternaire proche du monoïde plaxique, fondé sur le schéma $cba \equiv bca \equiv cab$. Un algorithme proche de l'algorithme de Schensted nous permet de caractériser les classes d'équivalence et d'exhiber une section du monoïde. Nous énonçons également une correspondance de Robinson-Schensted pour le monoïde Chinois avant de nous intéresser au calcul du cardinal de certaines classes. Ce travail a permis de développer de nouveaux outils combinatoires. Nous avons trouvé un plongement de chacune des classes d'équivalence dans la plus grande classe. La dernière partie de cet article présente l'étude des relations de conjugaison.

Abstract

This paper presents a combinatorial study of the Chinese monoid, a ternary monoid related to the plactic monoid and based on the rewritings $cba \equiv bca \equiv cab$. An algorithm similar to Schensted's algorithm yields a characterisation of the equivalence classes and a cross-section theorem. We also establish a Robinson-Schensted correspondence for the Chinese monoid before computing the order of specific Chinese classes. For this work, we had to develop some new combinatorial tools. Among other things we discovered an embedding of every equivalence class in the greatest one. Finally, the end of this paper is devoted to the study of conjugacy classes.

* `cassaign@litp.ibp.fr` Julien Cassaigne Université Paris 7 – LITP – IBP – 2, Place Jussieu - 75251 Paris Cedex 05

† `Marc.Espie@ens.fr` Marc Espie 9, rue du Pot de fer 75005 Paris

‡ `Florent.Hivert@ens.fr` Florent Hivert ENS, 45 rue d'Ulm 75005 Paris

§ `dk@litp.ibp.fr` Daniel Krob Université Paris 7 – LITP – IBP – 2, Place Jussieu - 75251 Paris Cedex 05

¶ `Jean-Christophe.Novelli@ens.fr` Jean-Christophe Novelli ENS, 45 rue d'Ulm 75005 Paris

Introduction

A large part of modern algebraic combinatorics can be traced back to Schensted's seminal paper of 1961 (cf [11]). In this paper, Schensted introduced an algorithm (now called Schensted's algorithm) for solving the elementary algorithmical problem of finding the length of the longest increasing subsequence of a sequence of integers.

Precisely, this algorithm associates with every word w over some totally ordered alphabet A a Young tableau $P(w)$, called the *insertion tableau* of w, and the length of the first row of $P(w)$ is exactly the length of the longest increasing subsequence of w. This property was generalized by Greene who gave an interpretation of the length of every row of the insertion tableau in terms of maximal lengths of shuffles of increasing subsequences (cf [3]). It is also worth noting that Schensted's algorithm is the main ingredient of the Robinson-Schensted correspondance which is a fundamental bijection between words and pairs (P, Q) of Young tableaux of the same shape (cf [10]).

In 1970, Knuth proved that the insertion tableaux of two words u, v are the same if and only if these words are equivalent with respect to the congruence of A^* generated by the relations

$$\begin{array}{ll} aba \equiv baa,\ bba \equiv bab & \text{when } a < b\ , \\ acb \equiv cab,\ bca \equiv bac & \text{when } a < b < c \end{array} \qquad (1)$$

(see [4]). Rather curiously Knuth did not use the fact that these relations were the definition relations of a monoid. This simple but fundamental remark was made by Lascoux and Schützenbeger who decided to call *plactic monoid* the monoid defined by the relations (1) (cf [6]). Initially they used the plactic monoid to derive one of the first proofs of the so-called Richardson-Littlewood rule (which yields a combinatorial interpretation of the multiplicity of a Schur function in a product of Schur functions, i.e., the multiplicity of a irreducible representation of $\mathfrak{S}_n$ of a tensor product of two irreducible representations of $\mathfrak{S}_n$) (cf [12, 9]). Lascoux and Schützenberger then developped in a decisive way the theory of the plactic monoid which became the main tool in several combinatorial contexts (Kostka-Foulkes polynomials, charge and cocharge of a Young tableau, standards bases, etc) (see [6, 7] for instance).

More recently Date, Jimbo and Miwa discovered a strong relationship between the usual Robinson-Schensted correspondance and the quantum group $U_q(gl_n)$ (cf [1]). They showed that if $V_{(1)}$ denotes the basic representation of $U_q(gl_n)$, the transition matrix in $V_{(1)}^{\otimes k}$ from the basis of monomial tensors (i.e. words) to the Gelfand-Zetlin basis (indexed by pairs of Young tableaux) specializes when $q = 0$ to a permutation matrix given by the Robinson-Schensted map. In the same spirit, Leclerc and Thibon even obtained a quantum characterization of the plactic monoid. They proved that two words $u = a_{i_1} \ldots a_{i_k}$ and $v = a_{j_1} \ldots a_{j_k}$ of $\{ a_1 < \ldots < a_n \}^*$ are equivalent with respect to the plactic relations iff one has

$$x_{i_1,i_1} \ldots x_{i_k,i_k} \equiv x_{j_1,j_1} \ldots x_{j_k,j_k} \bmod q\,\mathcal{L}$$

in the classical quantum deformation $A_q(Mat_n)$ of the coordinate ring of $n \times n$ matrices over a field K of characteristic 0, where $\mathcal{L}$ denotes the lattice generated over $K[q]$ by quantum bitableaux (cf [8] for details).

All these facts clearly motivate further study of the plactic monoid as a fundamental combinatorial and algebraic object. Several characterizations of the plactic monoid have been given (cf [6]), but none was based on growth properties. The Hilbert series of the plactic monoid is indeed given by Schur-Littlewood formula. In other words,

$$\sum_{I \in \mathbb{N}^n} p_I \, a^I = \frac{1}{\prod\limits_{a \in A} (1-a) \prod\limits_{a<b \in A} (1-ab)} \,, \tag{2}$$

where m_I denotes the number of plactic classes of evaluation I. It seems therefore natural to try to find all monoids whose Hilbert series are given by the above identity. Duchamp, Krob and Schützenberger obtained a partial result in this direction. They showed that there were exactly three families of ternary monoids (i.e., generated by 3-letters multi-homogeneous relations) whose Hilbert series is given by formula (2). But only two of these three families are regular: the plactic monoid, and the so-called Chinese monoid. This Chinese monoid appeared to be naturally connected with classical bijective proofs of Schur-Littlewood identities. Therefore, a deeper study of this new object seemed to be in order.

This paper constitutes a first combinatorial study of the chinese monoid, where we obtained several new results. After pointing out some basic properties of the Chinese monoid, we devote Section 2 to an insertion algorithm that yields a cross-section theorem for the Chinese monoid. Sections 3 and 4 contain a fine study of Chinese classes in the standard case: in violent contrast to the plactic monoid where the distribution of all words in different classes is rather uniform, there is a great class in the Chinese case. Section 5 gives a framework for the enumeration of Chinese classes. Finally Section 6 is devoted to the study of the conjugacy relation : we prove that, as in the plactic case, conjugacy classes are identical to evaluation classes.

ACKNOWLEDGEMENTS
We would like to thank Joël-Yann Fourré for his help.

1 Definition and first properties

1.1 Definition

Definition 1.1 (Duchamp, Krob, [2]) *Let $(A,<)$ be a totally ordered alphabet. The* Chinese congruence *is the congruence defined by the relations*

$$cba \equiv cab \equiv bca \qquad \text{for every } a<b<c, \tag{3}$$

$$aba \equiv baa\,,\ bba \equiv bab \qquad \text{for every } a<b. \tag{4}$$

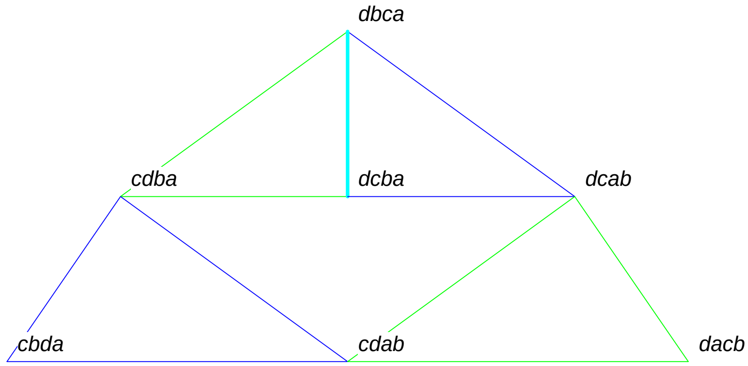
The Chinese monoid $CH(A,<)$ *is the quotient monoid of A^* by the Chinese congruence.*

Note 1.2 Obviously, (3) and (4) together are equivalent to

$$cba \equiv cab \equiv bca \qquad \text{for every } a \le b \le c. \tag{5}$$

Therefore we will almost exclusively use (5) throughout the paper.

For instance, Figure 1 shows the congruence class of $dcba$.



This is the congruence graph of *dcba*. Each edge stands for an elementary rewriting. The thick edge between *dcba* and *dbca* means that these words are equivalent thanks to *two* elementary congruences, namely *dcb* $\equiv$ *dbc* and *cba* $\equiv$ *bca*.

Figure 1: The class of *dcba*

1.2 Schützenberger's Involution

We shall denote by $CH(A, >)$ the Chinese monoid built over the alphabet A supplied with the opposite order of $<$. Denote by $(A^*)^\circ$ the opposite monoid associated with A^*. We consider the morphism $\natural$ from A^* into $(A^*)^\circ$ defined by:

$$\natural(a_1 \dots a_k) = a_k \dots a_1,$$

i.e., $\natural$ maps every word w to its mirror image. The morphism $\natural$ is compatible with the Chinese monoid structure, that is:

$$u \equiv_< v \quad \Leftrightarrow \quad \natural(u) \equiv_> \natural(v), \tag{6}$$

i.e., $\natural$ actually defines a morphism from $CH(A, <)$ into $(CH(A, >))^\circ$. To prove this, it is enough to check that

$$\natural(u) \equiv \natural(v)$$

for every pair (u, v) of words involved in an elementary congruence of $CH(A, <)$. Choose $a \leq b \leq c$ as in (5). We have

$$\natural(cba) = abc \quad \equiv_> \quad \natural(cab) = bac \quad \equiv_> \quad \natural(bca) = acb,$$

for the Chinese congruence $\equiv_>$ of $CH(A, >)$.

For an alphabet A over the ordered letters $a, b, \dots, z$ consider the elementary morphism $\mathcal{I}\colon (A^*, <) \to (A^*, >)$ defined by $\mathcal{I}(a) = z, \mathcal{I}(b) = y, \dots, \mathcal{I}(z) = a$. Then $\# = \mathcal{I} \circ \natural$ is an isomorphism from $CH(A, <)$ to $(CH(A, <))^\circ$, which is in fact an involution of the set $CH(A, <)$, called *Schützenberger's involution* of $CH(A, <)$.

As an elementary application, Schützenberger's involution helps finding isomorphic classes in the Chinese monoid. Figure 2 shows a simple example of classes that are Schützenberger-equivalent. Note also the symmetry of Figure 1, since the class of *dcba* is invariant under Schützenberger's involution.

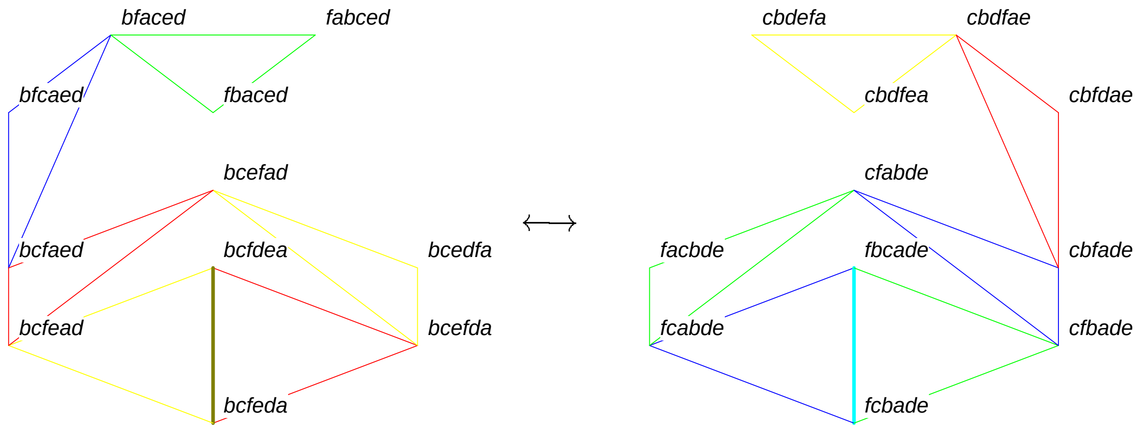


Figure 2: Schützenberger-equivalent classes: *facbde* and *bcedfa*

1.3 Standardization

Let w be a word over the alphabet A. We associate with it a standard word[1] $Std(w)$ over the alphabet $A \times \mathbb{N}$ obtained by numbering all occurrences of the same letter $1, 2, 3, \ldots$ from *right to left*. For instance, we have:

$$Std(\mathsf{babbaacb}) = \mathsf{b_4 a_3 b_3 b_2 a_2 a_1 c_1 b_1} \ .$$

This standardization process is compatible with the chinese congruence. Indeed:

$$u \equiv v \quad (CH(A, <)) \quad \Rightarrow \quad Std(u) \equiv Std(v) \quad (CH(A \times \mathbb{N}, <)), \tag{7}$$

where the order over $A \times \mathbb{N}$ is the lexicographic order defined by

$$(\alpha, i) < (\beta, j) \qquad \text{iff} \qquad \alpha < \beta \text{ or } (\alpha = \beta \text{ and } i < j).$$

1.4 Collapsing the intervals

Let I be an interval of A, i a letter of I and consider the new alphabet $A_I = A \setminus I \cup \{i\}$ that we totally order by restricting $<$ to $A \setminus I$ and defining for every $\alpha \in A \setminus I$

$$\alpha < i \qquad \text{iff} \qquad \alpha < \beta \text{ for some } \beta \in I.$$

This definition makes sense since I is an interval. We can now define the morphism χ_I from A^* into ${A_I}^*$ by setting

$$\chi_I(\alpha) = \begin{cases} \alpha & \text{if } \alpha \notin I \\ i & \text{if } \alpha \in I \end{cases} \ .$$

In other words, χ_I maps every word to the word obtained by replacing every letter of I with the same single letter i. This morphism is compatible with the Chinese congruence:

$$u \equiv v \quad \Rightarrow \quad \chi_I(u) \equiv \chi_I(v) \ . \tag{8}$$

[1] A standard word is a word without repetition of letters.

We just have to check property (8) for the defining relations of the Chinese congruence. Just observe that:

$$\chi_{\{a,b\}}(cba) \equiv \chi_{\{a,b\}}(cab) \equiv \chi_{\{a,b\}}(bca), \;\; \chi_{\{b,c\}}(cba) \equiv \chi_{\{b,c\}}(cab) \equiv \chi_{\{b,c\}}(bca).$$

But these last relations reduce to

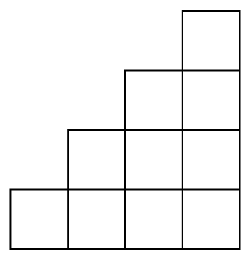
$$cbb \equiv cbb \equiv bcb, \;\; bba \equiv bab \equiv bba,$$

which is obviously satisfied. This concludes the proof of (8).

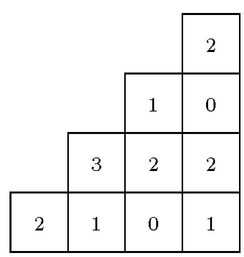
2 A representation of the Chinese monoid

2.1 Chinese Staircases

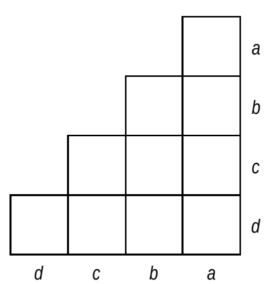
Let A be a totally ordered alphabet of n letters. A p-staircase diagram is a Ferrers diagram of shape $(1, 2, \dots, p)$ that we draw in the following way:



the example being given here for $p = 4$. A *partial Chinese staircase* is a p-staircase diagram filled with nonnegative integers[2] , e.g.,



We index the rows (resp. the columns) of the diagram with an initial segment of A from top to bottom (resp. from right to left.) i.e.,



this example being for $A = \{\mathsf{a}, \mathsf{b}, \mathsf{c}, \mathsf{d}, \mathsf{e}\}$, $p = 4$.

[2] Most of the time, we will omit zeros for clarity.

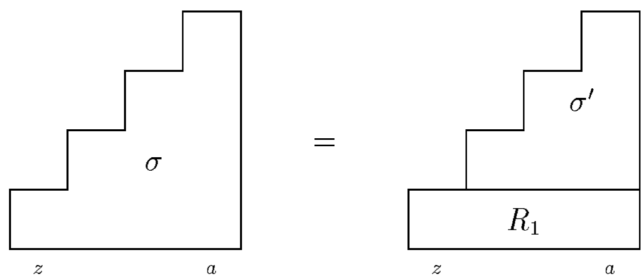
We denote indifferently by $\sigma_{\alpha\beta}$ the cell in row α, column β or its contents. Formally, a partial staircase is just an application σ from $A \times A$ into $\mathbb{N}$ such that $\sigma_{\alpha\beta} \neq 0$ implies $p \geq \alpha \geq \beta$. Furthermore we shorten $\sigma_{\alpha\alpha}$ into σ_α.

A *full Chinese staircase* is a n-Chinese staircase over a n letters alphabet. In this paper, 'Chinese staircase' or 'staircase' for short designates a full Chinese staircase unless otherwise precised.

Note 2.1 Strictly speaking, a given staircase is associated with an initial segment of the alphabet. However, it is fairly easy to extend a p-staircase to a $p+1$-staircase by setting the appropriate entries to 0. Conversely, if z is the greatest letter of a p-staircase σ, and if the last row of σ is empty, σ reduces to a $p-1$-staircase.

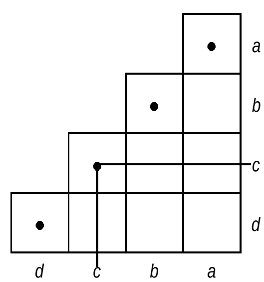
We need a form of structural induction on staircases. Let σ be a p-staircase over the initial segment $a, b, \ldots, y$. Let row R be an application from $a, b, \ldots, z$ into $\mathbb{N}$. We denote by $\sigma' = (\sigma,\ R)$ the $p+1$-staircase built from σ by setting $\sigma'_{z\alpha} = R(\alpha)$.

Conversely, if σ is a p-staircase with $p > 0$, let R_1 be the bottom row of σ, σ' the $p-1$-staircase obtained by deleting the bottom row of σ. Then $\sigma = (\sigma',\ R_1)$.



Definition 2.2 *The* hook *corresponding to a letter α is the union of the row and the column indexed by α.*

Example 2.3 The hook of c and the first diagonal of the staircase:

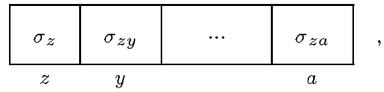


Note 2.4 The way we draw Ferrers diagrams is neither the French nor the English usual notation. However, since we are going to describe an insertion algorithm, this is a suitable notation to mimic the classical Schensted algorithm, where letters move from bottom to top and from right to left.

A word w is said to be a *Chinese row* of type z if and only if it has the following structure

$$w = (za)^{n_a} \ldots (zy)^{n_y}\, (z)^{n_z}$$

where $a, b, \ldots, z$ denotes the initial segment of A ending with z, and where every n_α belongs to $\mathbb{N}$. Now let σ be a Chinese staircase. We associate to every row of σ a Chinese row in a natural way. Indeed, if the z^{th} row of σ has the form



then the associated Chinese row is just the word equal to

$$(za)^{\sigma_{za}} \dots (zy)^{\sigma_{zy}}\,(z)^{\sigma_z}$$

We say that a word w is a *Chinese staircase word* if and only if it can be written as

$$w = l_a\, l_b \dots\, l_z \tag{9}$$

where the l_α are Chinese rows of respective increasing types $a, b, \dots, z$.

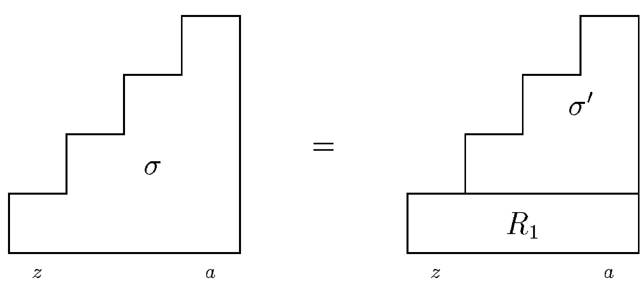
Definition 2.5 *The row-reading of a Chinese staircase σ is the word w obtained by concatenating all Chinese rows associated with the rows of σ from top to bottom.*

We will define later (in Definition (2.18)) the column-reading of a Chinese staircase as a dual notion.

2.2 The Insertion Algorithm

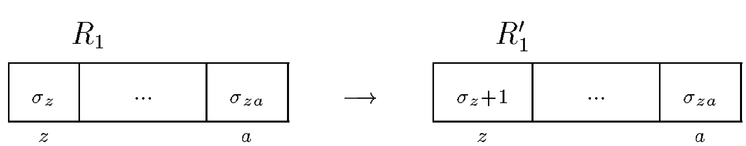
We shall now describe an algorithm that sends a word of A^* to a Chinese staircase. The basic idea is to try to adapt Schensted algorithm (Schensted, [11], Knuth, [4], Knuth, [5]). The basic step builds upon a Chinese staircase σ and a letter α a new Chinese staircase denoted by $\sigma.\alpha$. Hence, starting with the empty staircase ϵ, we build step by step a staircase $(\cdots((\epsilon.a_1).a_2)\cdots).a_k$ corresponding to the word $a_1 a_2 \dots a_k$.

Algorithm 2.6 (The insertion algorithm)
Let σ be a staircase, α a letter to insert in σ. Start with $\sigma = (\sigma',\ R_1)$, where R_1 is the bottom row of σ, z the greatest letter of σ.



1. If $\alpha > z$, then $\sigma.\alpha = \sigma$.

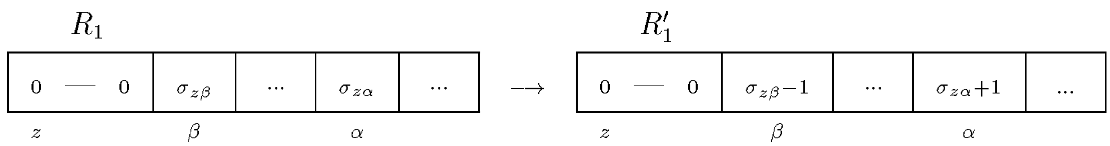
2. If $\alpha = z$, then $\sigma.\alpha = (\sigma',\ R_1')$ where R_1' is obtained from R_1 by adding 1 to cell σ_z:



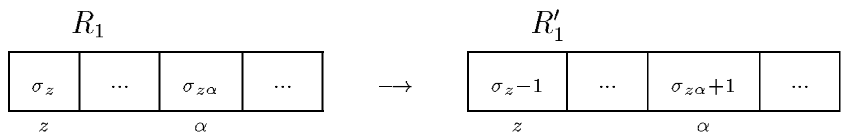
3. If $\alpha < z$, let β be the greatest letter whose cell on R_1 does not contain 0 or if such a β does not exist, set $\beta = \alpha$. Three distinct cases appear:

3*a*. If $\alpha \geq \beta$, then $\sigma.\alpha = (\sigma'.\alpha,\ R_1)$.

3*b*. If $\alpha < \beta < z$, then $\sigma.\alpha = (\sigma'.\beta,\ R_1')$, where R_1' is obtained from R_1 by adding 1 to cell $\sigma_{z\alpha}$ and substracting 1 from cell $\sigma_{z\beta}$:



3*c*. If $\alpha < \beta = z$, then $\sigma.\alpha = (\sigma',\ R_1')$, where R_1' is obtained from R_1 by adding 1 to cell $\sigma_{z\alpha}$ and substracting 1 from cell σ_z:



Examples 2.7 On the alphabet a, b, c, words *cba*, *cab*, *bca* map to the same staircase:

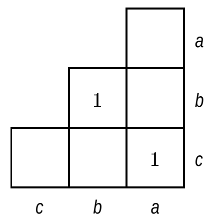


Figure 3 shows the insertion algorithm in action.

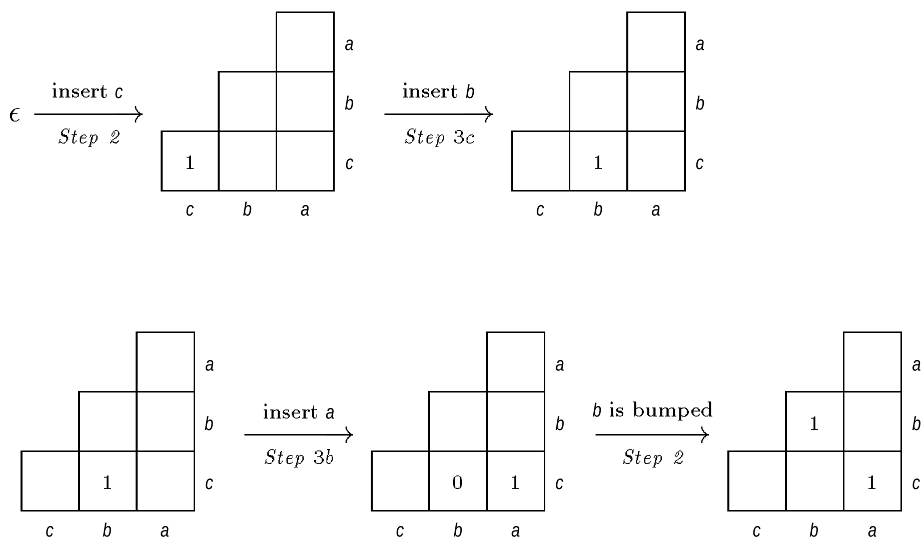
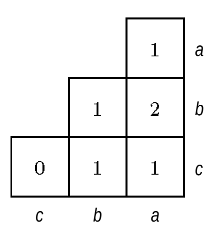


Figure 3: The insertion of *cba*

Similarly the reader may check that the word *abababcbca* gives:



We will now establish some basic properties of the insertion algorithm.

Definition 2.8 *Write down explicitly all the steps in the insertion of a letter $\alpha = \alpha_1$. The insertion algorithm deals with a decreasing sequence of partial staircases:*

$$\begin{aligned}\sigma.\alpha_1 =& (\sigma'.\alpha_2, R'_1) && \textit{(by step } s_1\textit{)} = \\ & ((\sigma''.\alpha_3, R'_2), R'_1) && \textit{(by step } s_2\textit{)} = \\ & (((\sigma^{(3)}.\alpha_4, R'_3), R'_2), R'_1) && \textit{(by step } s_3\textit{)} = \\ & \cdots \\ & = ((\cdots(\sigma^{(k-1)}.\alpha_k, R'_{k-1}), \ldots), R'_1) && \textit{(by step } s_{k-1}\textit{)} \\ & = ((\cdots(\sigma^{(k)}, R_k), \ldots), R'_1) && \textit{(by step } s_k\textit{)}.\end{aligned}$$

The sequence $\alpha_1 \xrightarrow{s_1} (R'_1, \alpha_2) \xrightarrow{s_2} (R'_2, \alpha_3) \ldots (R'_{k-1}, \alpha_k) \xrightarrow{s_k} R'_k$ describes accurately the execution of the algorithm.[3] *We call such a sequence an* insertion sequence.

Definition 2.9 *Let σ be a staircase. An* exposed entry *is a cell holding a non zero value such that all cells to its west and to its south-west are empty (that is: a $\sigma_{\alpha\beta}$ such that $\sigma_{\alpha\beta} > 0$, $\sigma_{\gamma\delta} = 0$ for $\gamma > \alpha$ and $\delta > \beta$, and $\sigma_{\alpha\delta} = 0$ for $\delta > \alpha$). An* exposed letter *is a letter that indexes a column corresponding to an exposed entry.*

Proposition 2.10 *The insertion algorithm yields an increasing sequence of letters $\alpha_1, \alpha_2, \ldots \alpha_k$ that takes as values all the exposed letters that are greater than α_1. Moreover, changes of value occur on the rows of corresponding exposed entries, by application of Step 3b. Other intermediate steps correspond to Step 3a. The last step $(R'_{k-1}, \alpha_k) \xrightarrow{s_k} R'_k$ is either Step 2 or Step 3c; specifically, α_k is the greatest letter of row R'_k.*

Proof — Consider $\alpha_i \xrightarrow{s_i} \alpha_{i+1}$. Step s_i has to be Step $3a$ or Step $3b$. If s_i is Step $3a$, then $\alpha_{i+1} = \alpha_i$. If s_i is Step $3b$, then $\alpha_{i+1} = \beta > \alpha_i$. The letter β has to be an exposed letter; if it were not, there would an exposed letter γ corresponding to an entry to the lower left of β, and there would be a step $j < i$ in the insertion sequence that would read: $\alpha_j \xrightarrow{3b} \gamma$.

Step 1 ensures that the algorithm is well-defined even on partial staircases, but it can never occur for an insertion in a full staircase: by inspection of Step $3a$ and Step $3b$, the letter α_i always occur in $\sigma^{(i-1)}$. If α_i is the greatest letter of $\sigma^{(i-1)}$, then $i = k$ and the algorithm terminates by applying Step 2 or Step $3c$. □

Definition 2.11 *Let Σ be the set of Chinese staircases over A. The insertion algorithm defines an action $\mathcal{A}$ of the monoid A^* on Σ:*

$$\begin{array}{llcl}\mathcal{A}: & \Sigma \times A^* & \longrightarrow & \Sigma \\ & (\sigma, w) & \longmapsto & \sigma.w = (\cdots((\sigma.a_1).a_2)\cdots).a_k,\end{array}$$

Looking for analogies with Schensted algorithm, it seems natural to try and find what a good Q-symbol might be in this context. This will help us find a kindred of Robinson-Schensted correspondence. This question we will address in Section 5.2.

Definition 2.12 *We denote by $C(\sigma)$ the set of words w of A^* such that $\epsilon.w = \sigma$. We also denote by $C(w)$ the Chinese classe of w. Theorem 2.14, to follow, fully justifies the use of a similar notation.*

[3] In semantics, this is called a trace of the algorithm.

Lemma 2.13 *For any two words of $C(\sigma)$, one is a permutation of the other.*

Proof — Set

$$|\sigma|_\alpha = \sum_{\alpha>\beta} \sigma_{\alpha\beta} + \sum_{\beta>\alpha} \sigma_{\beta\alpha} + \sigma_\alpha,$$

that is: sum all the cells in the hook of α. It is immediate to check that $|\sigma.\alpha|_\beta = |\sigma|_\beta + \delta_\alpha^\beta$, where δ_α^β denotes the Kronecker symbol of α and β. Hence, if $\sigma = \epsilon.w$, $|\sigma|_\alpha$ counts the number of α that have been inserted in ϵ, i.e., the number of α in w, $|w|_\alpha$. □

2.3 The Cross-Section Theorem

Theorem 2.14 *The Chinese staircase words form a cross-section of the Chinese monoid. More precisely:*

- *Property 1: For any words v and w for which the insertion algorithm yields the same staircase σ, v and w are equivalent under the Chinese congruence.*
- *Property 2: For any words v and w equivalent under the Chinese congruence, for any staircase σ, $\sigma.v = \sigma.w$.*

Proof —
Property 1: First we prove that we can assume that v is the row normal form of σ.

Lemma 2.15 *Let σ be a staircase and t the associated row normal form. Then σ is exactly the staircase obtained by the insertion algorithm applied to t.*

Proof — This lemma is obvious since the rows of σ are filled from top to bottom when inserting t. □

Hence, let w be a generic word and $v = t$ be the row normal form of σ. The proof is obtained by induction on the number of letters of w. If $|w| = 2$, there is nothing to prove. If $|w| \geq 3$, decompose $w = w_1\alpha$ with $\alpha \in A$. Let Property 1 hold for w_1: if σ_1 is the staircase obtained by the insertion of w_1, its row normal form t_1 is congruent to w_1. We are going to "simulate" the insertion algorithm through elementary congruences, proving thus that $w_1\alpha$ can actually be rewritten into t in an effective way. Paralleling the algorithm, we perform a structural induction on the staircase σ_1.

Let σ be a partial staircase of σ_1, let t be the row normal form of σ. Decompose $\sigma = (\sigma',\ R_1)$ and $t = t'r$ with r the Chinese row associated with R_1. The induction hypothesis states that, for any β smaller than the greatest letter of σ', t' concatenated with β is congruent to the row normal form of $\sigma'.\beta$. Keep in mind that $r = (za)^*(zb)^* \dots (zy)^*z^*$ and insert α in r:

- Step 1: Not applicable.
- Step 2: $\sigma.\alpha = (\sigma',\ R_1')$, with $r' = rz$. No rewriting is necessary.
- Step 3a: By applying the insertion algorithm, we obtain $\sigma.\alpha = (\sigma'.\alpha,\ R_1')$. Since $\beta \leq \alpha < z$, $r = (za)^*(zb)^* \dots (zy)^*$. Denote by r' the Chinese row associated with R_1'. The induction hypothesis applies since $\alpha < z$, so we just have to verify

that $r\alpha \equiv \alpha r'$. Since $\alpha \geq \beta$, for every $(z\gamma)$ occuring in r, $\gamma \leq \beta \leq \alpha$, so that $(z\gamma)\alpha \equiv \alpha(z\gamma)$: α crosses through each pair of letters $(z\gamma)$. Hence $r\alpha \equiv \alpha r'$, where r' is the Chinese word associated with R_1'.

- Step $3b$: By applying the insertion algorithm, we obtain similarly $\sigma.\alpha = (\sigma'.\beta,\ R_1')$. Since $\beta < z$, $r = (za)^*(zb)^*\ldots(zy)^*$. The induction hypothesis applies since $\beta < z$, so we just have to verify that $r\alpha \equiv \beta r'$.
  - Since $\alpha < \beta$, $(z\beta)\alpha \equiv \beta(z\alpha)$, hence $(z\beta)$ is replaced with $(z\alpha)$.
  - Since β is defined as the *greatest* letter whose cell does not contain 0, for every $(z\gamma)$ occuring in r, $(z\gamma)\alpha \equiv \alpha(z\gamma)$: β crosses through each pair of letters $(z\gamma)$, as in Step $3a$.
  - For every $(z\gamma)$ occuring in r with $\gamma > \alpha$, $(z\gamma)(z\alpha) \equiv z(z\alpha)\gamma \equiv (z\alpha)(z\gamma)$, so that $(z\alpha)$ is able to move to the correct position in r',
- Step $3c$: Let us write $r = (za)^*(zb)^*\ldots(zy)^*z^k$ with $k > 0$. Since $\alpha < z$, $z^k\alpha \equiv (z\alpha)z^{k-1}$. Proceed as in Step $3b$ to move $(z\alpha)$ to the correct position and conclude that $r\alpha \equiv r'$.

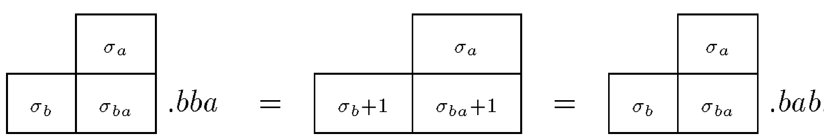
Property 2: We have to check that the insertion algorithm is compatible with the Chinese congruence. Clearly, it is enough to check Property 2 for elementary rewritings. So we must prove:

$$\sigma.cba = \sigma.bca = \sigma.cab,$$

for every $a \leq b \leq c$. The proof proceeds by induction on the size p of the staircase. The induction hypothesis is

> *Let σ be a p-staircase of greatest letter z, let a, b, c be three letters such that $a \leq b \leq c \leq z$. Then $\sigma.cba = \sigma.cab = \sigma.bca$.*

If $p = 2$, checking that both insertions give the same result is easy. For instance,



Lest σ be a p-staircase of greatest letter z. Let a, b, c be three letters such that $a \leq b \leq c \leq z$. Assume that the induction hypothesis holds for $p - 1$. Decompose σ as $\sigma = (\sigma',\ R)$ and consider all possible cases.

1. $c \neq z$. So the induction hypothesis holds for every triplet of suitable letters smaller than c or equal to c.

   1.1. Row R contains no exposed letter, or the exposed letter d in row R verifies $d \leq a$. Then the insertion sequences of a, b, and c do not interfere with R, that is: $a \xrightarrow{3a} (R, a)$, $b \xrightarrow{3a} (R, b)$, and $c \xrightarrow{3a} (R, c)$. By induction $\sigma'.cba = \sigma'.cab = \sigma'.bca$, hence $\sigma.cba = \sigma.cab = \sigma.bca$.

1.2. The exposed letter in row R is a' with $a < a' \le b$. Then the insertion sequence of a starts with $a \xrightarrow{3b} (R', a')$, whereas b and c yield $b \xrightarrow{3a} b$ and $c \xrightarrow{3a} c$. So we have $\sigma.cba = (\sigma'.cba',\ R')$, $\sigma.bca = (\sigma'.bca',\ R')$, $\sigma.cab = (\sigma'.ca'b,\ R')$. The induction hypothesis applies for σ', a', b, c, and the result holds. The following diagram describes the transformation from R to R':

0 — 0	$\sigma_{za'}$	$\cdots$	σ_{za}	$\cdots$	R
0 — 0	$\sigma_{za'}-1$	$\cdots$	$\sigma_{za}+1$	$\cdots$	R'
z b	a'		a		

1.3. The exposed letter in R is b' with $b < b' \le c$. We still have that $c \xrightarrow{3a} c$, so we just have to examine what happens for the insertion of ba versus the insertion of ab.

First we insert ba in σ. The start of the insertion sequence of b is: $b \xrightarrow{3b} (R_1, b')$. Let a' be the exposed letter of R_1. We necessarily have $b' \ge a' \ge b$. Therefore, the subsequent insertion of a yields $a \xrightarrow{3b} (R', a')$.

Now we insert ab in σ. The start of the insertion sequence of a is: $a \xrightarrow{3b} (R_2, b')$. Let d be the exposed letter of R_2. If $d \ge b$, then $d = a'$ and $b \xrightarrow{3b} (R', a')$. If $d < b$, then $b = a'$ and $b \xrightarrow{3b} (R', a')$. Thus $\sigma.cba = \sigma.cab = (\sigma'.cb'a',\ R')$, and $\sigma.bca = (\sigma'.b'ca',\ R')$. The induction hypothesis applies for σ', a', b', c, and the result holds. The following diagram describes the transformation from R to R':

0 — 0	$\sigma_{zb'}$	$\cdots$	$\sigma_{za'}$	$\cdots$	σ_{zb}	$\cdots$	σ_{za}	$\cdots$	R
0 — 0	$\sigma_{zb'}-1$	$\cdots$	$\sigma_{za'}-1$	$\cdots$	$\sigma_{zb}+1$	$\cdots$	$\sigma_{za}+1$	$\cdots$	R'
z	b'		a'		b		a		

1.4. The exposed letter in R is c' with $c \le c'$.

First we insert cba in σ. the start of the insertion sequence of c is: $c \xrightarrow{3b} (R_1, c')$ or $c \xrightarrow{3c} (R_1, c')$. Let b' be the exposed letter of R_1. We necessarily have $c' \ge b' \ge c$. Write the start of the insertion sequence of b: $b \xrightarrow{3b} (R_2, b')$ or $b \xrightarrow{3c} (R_2, b')$. Let a' be the exposed letter of R_2. We necessarily have $b' \ge a' \ge b$. Write the start of the insertion sequence of a: $a \xrightarrow{3b} (R', a')$ or $a \xrightarrow{3c} (R', a')$.

Examining the insertion of bca and cab in σ is tedious, and identical to the previous case considerations. In the end, we have that $\sigma.cba = \sigma.cab = \sigma.bca = (\sigma'.c'b'a',\ R')$, so the result holds.

2. $c = z$. In that case, the insertion of c always proceeds as $c \xrightarrow{2} R_1$, with c an exposed letter of R_1.

2.1 Row R contains no exposed letter, or the exposed letter d in row R verifies $d \le b$. Then $\sigma.cba = \sigma.cab = \sigma.bca = (\sigma'.b,\ R')$. The result holds. The following diagram describes the transformation from R to R':

0 — 0	σ_{cb}	$\cdots$	σ_{ca}	$\cdots$	R
0 — 0	σ_{cb}	$\cdots$	$\sigma_{ca}{+}1$	$\cdots$	R'
c	b		a		

2.2. The exposed letter in R is b' with $c \geq b' > b$. We may have $c = b'$ or $c > b'$, but this is irrelevant; in any case, we have $\sigma.cba = \sigma.cab = \sigma.bca = (\sigma'.b',\ R')$. The result holds. The following diagram describes the transformation from R to R':

— 0	$\sigma_{cb'}$	$\cdots$	σ_{cb}	$\cdots$	σ_{ca}	$\cdots$	R
— 0	$\sigma_{cb'}{-}1$	$\cdots$	$\sigma_{cb}{+}1$	$\cdots$	$\sigma_{ca}{+}1$	$\cdots$	R'
	b'		b		a		

Thus ends the proof of Property 2 and of the theorem. □

Note 2.16 Let σ be a staircase, v, w two words. By Property 2, if $v \equiv w$, $\sigma.v = \sigma.w$. So action $\mathcal{A}$ is compatible with the Chinese congruence, and the quotient $\mathcal{A}/_{\equiv}$ is well defined. Consider σ_1 another staircase. By Theorem 2.14, $C(\sigma_1)$ is a Chinese class, so the staircase $\sigma.\sigma_1 = \sigma.w$ is constant for $w \in C(\sigma_1)$.[4] This law defines an action of Σ on itself which is isomorphic to $\mathcal{A}/_{\equiv}$ thanks to the isomorphism $\sigma \to C(\sigma)$. Finally, $(\Sigma, .)$ is isomorphic to the Chinese monoid $Ch(A, <)$.

Note 2.17 We can easily prove that Property 2 implies the following fact:

– Property 3: For any t and t' staircase words, $t \equiv t'$ implies $t = t'$.

Consider a word w. Let $\sigma = \epsilon.w$. The row normal form of σ is a normal form of $C(w)$: it is identical for all words of $C(w)$ and uniquely identifies σ, therefore $C(w)$.

2.4 Dual properties

We define the column-reading of a staircase the way we defined the row-reading in Section 2.1. A word w is a *Chinese column* of type a if and only if it has the following structure

$$w = (a)^{n_a} \ldots (ba)^{n_b}\,(za)^{n_z}$$

where $a, b, \ldots, z$ denotes the final interval of A beginning with a and where every n_α belongs to $\mathbb{N}$. We can also define the Chinese column associated with the column of a given staircase σ (we read the column from top to bottom.)

We say that a word w is a Chinese column staircase word if and only if it can be written as

$$w = l'_a\, l'_b \ldots\, l'_z \tag{10}$$

where the l'_α are Chinese columns of respective types α.

[4] For a completely algorithmic definition, use t_1, the row normal form of σ_1, in the insertion algorithm.

Definition 2.18 *The column normal form of a Chinese staircase σ is the word w obtained by concatening all Chinese columns associated with the columns of σ from right to left.*

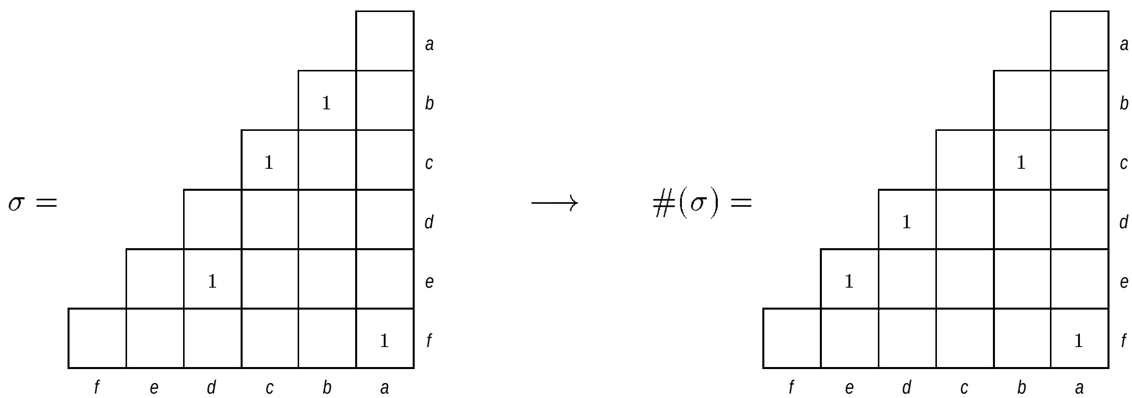
Proposition 2.19 *Let σ be a staircase and t (resp. c) the Chinese row (resp. column) staircase word associated with σ. Then t and c are equivalent with respect to the Chinese congruence.*

Proof — It is clear that σ is the staircase obtained by the insertion algorithm applied to c. The conclusion then follows from Property 1.

The following result gives us a simple geometric interpretation of Schützenberger's involution on staircases. Define the second diagonal of a staircase σ over the alphabet $A = \{a, b, \ldots, z\}$ as the line of cells indexed by $(a, z), (b, y), \ldots$.

Proposition 2.20 *Let σ be a staircase. Schützenberger's involution acts on σ by performing a symmetry about the second diagonal of σ.*

Example 2.21



The reader may be interested to note that this is the same example as in Figure 2.

3 Backtracking the Insertion Algorithm

We now place consider the standard case, where all letters used in a given word are distinct.

Definition 3.1 *Let σ be a staircase. σ is said to be a* standard staircase *if a word of $C(\sigma)$ is a permutation over letters of the alphabet, σ is a* full standard staircase *if σ is a permutation over all letters of the alphabet. Thanks to Lemma 2.13, we know that σ is a standard staircase if and only if each hook of σ contains at most one 1, all other entries being 0; σ is a full standard staircase if and only if each hook of σ contains exactly one 1.*

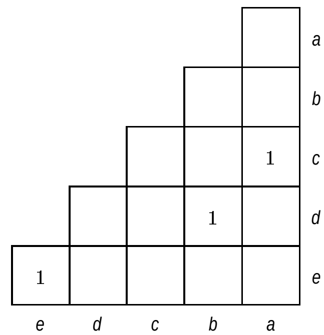
In that case, we can identify non null entries and letters of the alphabet. We speak of 'the letter α' in a staircase as a shorthand notation for 'the non null entry corresponding to α'.

3.1 Link Representation

Definition 3.2 *Let σ be a standard staircase. Use σ to define a partial involution ρ of A: for every non-empty $\sigma_{\alpha\beta}$, let ρ exchange α and β.*

The link representation $\lambda(\sigma)$ *of σ is a representation of ρ that we obtain as follows: dispose all letters involved in σ in lexicographic order and link two letters whenever they appear together in σ. Link a letter on the diagonal with itself. For instance,*

if $\sigma =$



, *then* $\lambda(\sigma) = \mathsf{abcde}.$

Obviously, the link representation $\lambda(\sigma)$, the corresponding standard staircase σ and the corresponding involution ρ are equivalent representations, the link representation being more compact. For instance, recalling Figure 1 and Figure 2:

$$\lambda(\mathsf{cbda}) = \mathsf{abcd}, \quad \lambda(\mathsf{bcedfa}) = \mathsf{abcdef}, \quad \lambda(\mathsf{cbdefa}) = \mathsf{abcdef}.$$

Definition 3.3 *Let σ be a standard staircase, ρ the corresponding involution.*
A great letter *α verifies $\rho(\alpha) < \alpha$, a* small letter *α verifies $\rho(\alpha) > \alpha$, and a* neutral letter *α verifies $\rho(\alpha) = \alpha$.*
In the staircase representation, the great letters index rows that contain a 1, the small letters index columns that contain a 1, and the neutral letters occur on the diagonal.

Example 3.4 Take the class of abdc, of link representation abcd. That is, a and b are neutral, c is small, d is great.

3.2 A Converse of the Insertion Algorithm

In order to find all the words of $C(\sigma)$, we need to find back all the staircases that can occur in the sequence $\epsilon.a_1.\dots.a_k$.

Definition 3.5 *Let σ be a staircase. A* deletable entry *is a cell holding a non zero value such that all cells to its south-west are empty. A* deletable letter *is a letter that indexes a column corresponding to a deletable entry.*

Please note that, in the case of a standard staircase, exposed entries as defined in Definition 2.9 and deletable entries are identical.

Algorithm 3.6 (Converse of the insertion algorithm 2.6)
Let σ be a staircase, $\sigma_{\gamma\delta}$ a deletable entry of σ. The algorithm defines a set of rewriting rules $\sigma \underset{\delta}{\longrightarrow} \tau$ as follows:

- Rule 1: if δ is on the diagonal of σ, τ is the staircase derived from σ by substracting one from the deletable entry σ_δ.

- Rule $1'$: if δ is not on the diagonal of σ, τ is the staircase derived from σ by substracting one from the deletable entry $\sigma_{\gamma\delta}$ and putting a 1 on the diagonal in σ_γ.
- Rule $2_{\tilde\delta}$: let $\tilde\delta$ be a deletable letter such that $\tilde\delta > \delta$. This implies that $\tilde\delta$ is higher than δ in σ, and that the deletable entry $\sigma_{\gamma\delta}$ verifies $\gamma > \delta$. Let $\tilde\sigma$ be the staircase obtained from σ by removing all the rows under row γ, inclusive. Choose $\tilde\tau$ recursively by $\tilde\sigma \underset{\tilde\delta}{\longrightarrow} \tilde\tau$, and build τ by adding at the bottom of $\tilde\tau$, first the row obtained by putting a 1 in cell $\sigma_{\gamma\tilde\delta}$ and substracting 1 from the cell $\sigma_{\gamma\delta}$, then the rows of σ of indices greater than γ.

  Formally, $\tilde\sigma$ is defined over $A' = \{\alpha \in A \mid \quad \alpha < \gamma\}$ by $\tilde\sigma_{\alpha\beta} = \sigma_{\alpha\beta}$ for any $\alpha < \gamma$, any β. And similarly: $\tau_{\alpha\beta} = \tilde\tau_{\alpha\beta}$ for any $\alpha < \gamma$, any β; $\tau_{\gamma\beta} = \sigma_{\gamma\beta}$ for any β but δ, $\tilde\delta$; $\tau_{\gamma\delta} = \sigma_{\gamma\delta} - 1$; $\tau_{\gamma\tilde\delta} = 1$; $\tau_{\alpha\beta} = \sigma_{\alpha\beta}$ for $\alpha = \gamma$, any β but δ, $\tilde\delta$; $\tau_{\alpha\beta} = \sigma_{\alpha\beta}$ for any $\alpha > \gamma$, any β.

  Figure 4 shows a graphic explanation of what is going on.

We define the set $(\sigma \underset{\delta}{\longrightarrow} .)$ as follows:

$$\begin{array}{lll} (\sigma \underset{\delta}{\longrightarrow} .) & = \{\tau \in \Sigma \mid \quad \sigma \underset{\delta}{\longrightarrow} \tau\} & \text{if } \delta \text{ is a deletable letter of } \sigma, \\ (\sigma \underset{\alpha}{\longrightarrow} .) & = \emptyset & \text{if } \alpha \text{ is not a deletable letter of } \sigma. \end{array}$$

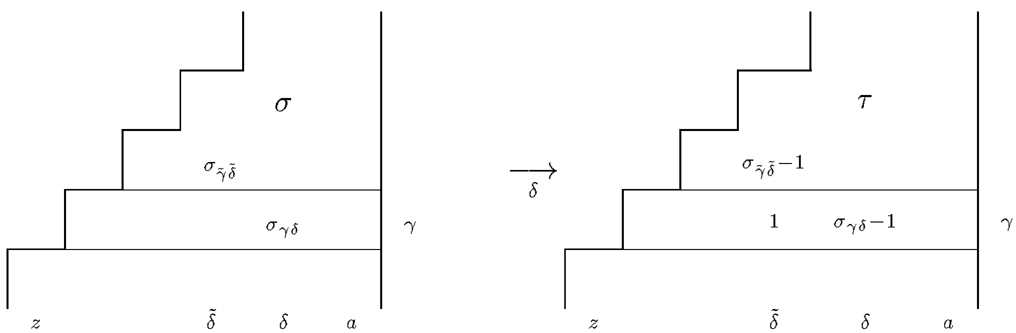


Figure 4: How Rule $2_{\tilde\delta}$ works.

Example 3.7 Consider the Chinese row staircase word *cbdfega* We display in Figure 5 the staircases obtained from *cbdfega* by applying Algorithm 3.6.

We now prove that this algorithm is correct.

Theorem 3.8 *Let σ be a standard staircase over n letters. Define:*

$$\begin{array}{rcl} \Delta(\sigma) & = & \{\delta \in A \mid \quad \exists\, \sigma' \in \Sigma',\ \sigma = \sigma'.\delta\}\,, \\ \sigma.\alpha^{-1} & = & \{\sigma' \in \Sigma \mid \quad \exists\, \alpha \in A,\ \sigma = \sigma'.\alpha\}\,. \end{array}$$

Then

- *$\Delta(\sigma)$ is the set of deletable letters.*
- *$\sigma.\alpha^{-1}$ is equal to $(\sigma \underset{\alpha}{\longrightarrow} .)$ for any α.*

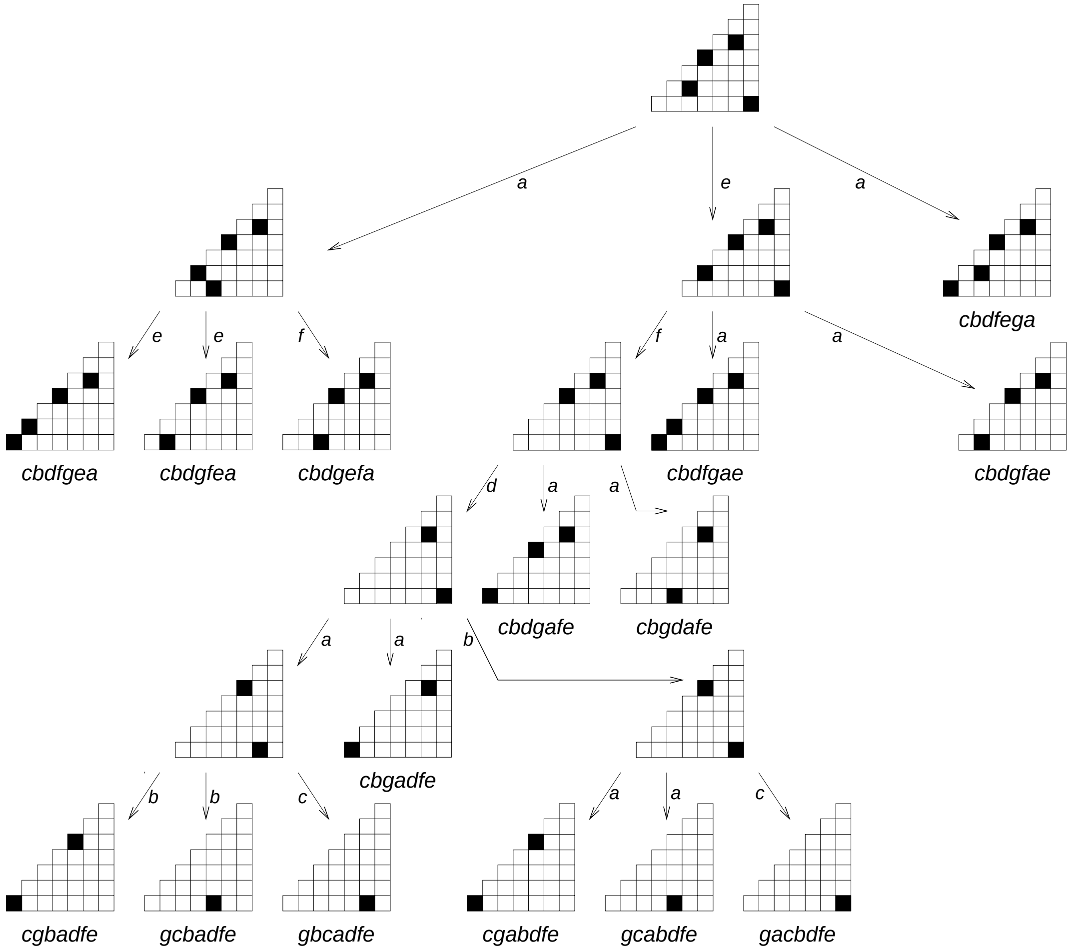


Figure 5: The cardinal of the class of *cbdfega* is 15.

The proof subdivides into two lemmas:

Lemma 3.9 $\Delta(\sigma)$ *is the set of deletable letters.*

Proof — If δ is a deletable letter, applying Rule 1 or Rule 1′ to σ results in a staircase τ that readily verifies $\tau.\delta = \sigma$. Conversely, let τ be a staircase, α a letter such that $\tau.\alpha = \sigma$. Consider the insertion sequence of α. At the first exposed entry $\sigma_{\gamma\beta}$ to the left of the column of α, the step of the insertion sequence reads $\alpha \xrightarrow{3b} \beta$ or $\alpha \xrightarrow{3c} \beta$. This step creates a deletable entry in $\sigma_{\gamma\alpha}$.[5] If there is no exposed entry to the left of the column of α, the insertion sequence ends by applying Step 2 and creates a deletable entry in σ_α. □

Lemma 3.10 $(\sigma \underset{\alpha}{\longrightarrow} .)$ *and* $\sigma.\alpha^{-1}$ *are the same for every* $\alpha \in \Delta(\sigma)$*.*

Proof — The proof proceeds by induction on the size of the staircase.

First, we prove that $(\sigma \underset{\alpha}{\longrightarrow} .) \subset \sigma.\alpha^{-1}$. In fact, we have already shown in Lemma 3.9 the case of Rule 1 and of Rule 1′. Assume the notations of Rule $2_{\bar{\delta}}$. Thanks to the

[5] In the standard case, the exposed entry $\sigma_{\gamma\beta} - 1$ vanishes: this explains why exposed entries and deletable entries are identical in this case.

induction hypothesis, $\tilde{\tau}.\tilde{\delta} = \tilde{\sigma}$. A close examination of Step 3b and Step 3c proves that $\tau.\delta = \sigma$.

Conversely, consider the insertion sequence that computes $\sigma.\alpha$. If the sequence of letters $\alpha_1, \ldots, \alpha_k$ is constant, if the last step is Step 2, then $\sigma.\alpha \underset{\alpha}{\longrightarrow} \sigma$ by Rule 1; if the last step is Step 3c, then $\sigma.\alpha \underset{\alpha}{\longrightarrow} \sigma$ by Rule 1$'$. If the insertion sequence $\alpha_1, \ldots, \alpha_k$ is not constant, the insertion sequence has to contain a step $\alpha = \alpha_i \overset{3b}{\longrightarrow} \alpha_{i+1} = \beta$. By the induction hypothesis, $\sigma^{(i)}.\beta \underset{\beta}{\longrightarrow} \sigma^{(i)}$, and therefore, $\sigma.\alpha \underset{\alpha}{\longrightarrow} \sigma$ by Rule 2_β. □

3.3 Applications

Theorem 3.11 *In the standard case, every class has an odd order.*

Proof — $C(\sigma.A^{-1})$ is a disjoint union of Chinese classes over $n-1$ letters. For $\sigma' \in \sigma.A^{-1}$, adding to an element of $C(\sigma')$ the letter δ such that $\sigma'.\delta = \sigma$ induces a bijection between $C(\sigma)$ and $C(\sigma.A^{-1})$. Since by induction every standard class over $n-1$ letters is odd, we have to prove that the order of $\sigma.A^{-1}$ is odd, otherwise the order of $C(\sigma.A^{-1})$, as an even sum of odd numbers, would be even.

We want to count all the staircases over $n-1$ letters that can give σ when inserting the missing letter.

$$\left|\sigma.A^{-1}\right| = \sum_{\delta \in \Delta(\sigma)} \left|\sigma.\delta^{-1}\right|$$

We now use the converse of the insertion algorithm since it enumerates all the different staircases that lead to a given staircase Note that as we consider a lower deletable entry in the staircase, the number of possibilities increase.

At this point, we strongly suggest that the reader look at Figure 5 while reading the end of the proof.

We now compute the cardinality of $\sigma.\delta^{-1}$ by induction. Since we are in the standard case, there is at most one deletable entry per row. If δ is the greatest deletable letter of σ, $|\sigma.\delta^{-1}| = 1$, since Rule $2_{\tilde{\delta}}$ does not apply. Else we may apply Rule 1$'$ or Rule $2_{\tilde{\delta}}$, for any $\delta' > \delta$. Rule 1$'$ (δ moving to the diagonal) accounts for one staircase. Rule $2_{\tilde{\delta}}$ (δ moving under a greater deletable element δ') accounts for $\left|\sigma.\delta'^{-1}\right|$, that is:

$$\left|\sigma.\delta^{-1}\right| = 1 + \sum_{\delta' > \delta} \left|\sigma.\delta'^{-1}\right|$$

Hence, if δ is the k^{th} deletable letter the order of $\sigma.\delta^{-1}$ is 2^{k-1}. Finally, the order of $\sigma.A^{-1}$ depends only upon $|\Delta(\sigma)|$:

$$\left|\sigma.A^{-1}\right| = 2^{|\Delta(\sigma)|} - 1.$$

□

Proposition 3.12 *The row normal form of a staircase σ is the minimal word in the class of σ for the lexicographic order.*

Proof — Thanks to standardization, we need only study the standard case. Define a total ordering on staircases: $\sigma < \sigma'$ if and only if the corresponding row normal forms t

and t' satisfy $t < t'$. Using structural induction we are going to show that, for $\sigma < \sigma'$, the smallest element in $C(\sigma)$ is smaller than the smallest element in $C(\sigma')$. Thanks to Theorem 3.8, we know that each staircase can be obtained by removing a deletable element at each step. But removing an element results in a staircase over $n-1$ letters, for which the induction hypothesis applies. Define $\sigma_\perp$ to be the staircase obtained by applying Rule 1 or Rule 1′ to σ for the highest deletable element (depending whether the highest letter is on the diagonal or not).

Lemma 3.13 *$\sigma_\perp$ is the smallest staircase of $\sigma.A^{-1}$.*

Proof — Let σ' be a staircase obtained by applying the rewriting rules to σ. Let d be the smallest row that differs between σ' and σ. Then, if d is the lowest non zero row of σ, $\sigma' = \sigma_\perp$ and $\sigma_\perp < \sigma$. If it is not the case, it is not very difficult to see that $\sigma' > \sigma$. Hence $\sigma_\perp < \sigma'$. □

We can then deduce that the smallest element is obtained by removing letters in each row from bottom to top and in a row from left to right (in the non standard case.) We thus obtain the row normal form of σ. □

4 The Great Class

We devote this section to several counting results concerning standard Chinese classes. This leads us to the construction of a bijection between the words of a given class and the so called Dyck words. We handle here the case of the largest class (the other cases may be found in the next section.) In that case, an integer associated to a Dyck word appears naturally. We call this integer the *weight* of the Dyck word.

4.1 The Great Class

Definition 4.1 *Let ω denote the maximal standard word over an alphabet A on n letters (for the lexicographic order): $\omega = zy\dots ba$. The* great class $Gr(n) = Gr(A)$ *is the Chinese class of ω.*

Note 4.2 Let $p = [\frac{n}{2}]$. In $Gr(n)$, the great letters are the p greatest letters of the alphabet and the small letters are the p smallest. If n is odd the $(p+1)^{\text{th}}$ letter is neutral, if n is even, there is no neutral letter. The i^{th} letter is associated with the $(n-i)^{\text{th}}$ letter. The non-zero entries occur precisely along the second diagonal of σ and are filled with 1.

Example 4.3 $Gr(6)$ is drawn in Figure 6.

In the standard case, non null entries correspond to neutral letters, or to pairs of a great and a small letter. Let us see what may happen to a given letter through the insertion algorithm.

By convention, a letter to be inserted, α, is initially small.

- If $\alpha \xrightarrow{3a} \alpha$, nothing much happens. α does not change of status and climbs up in the staircase.

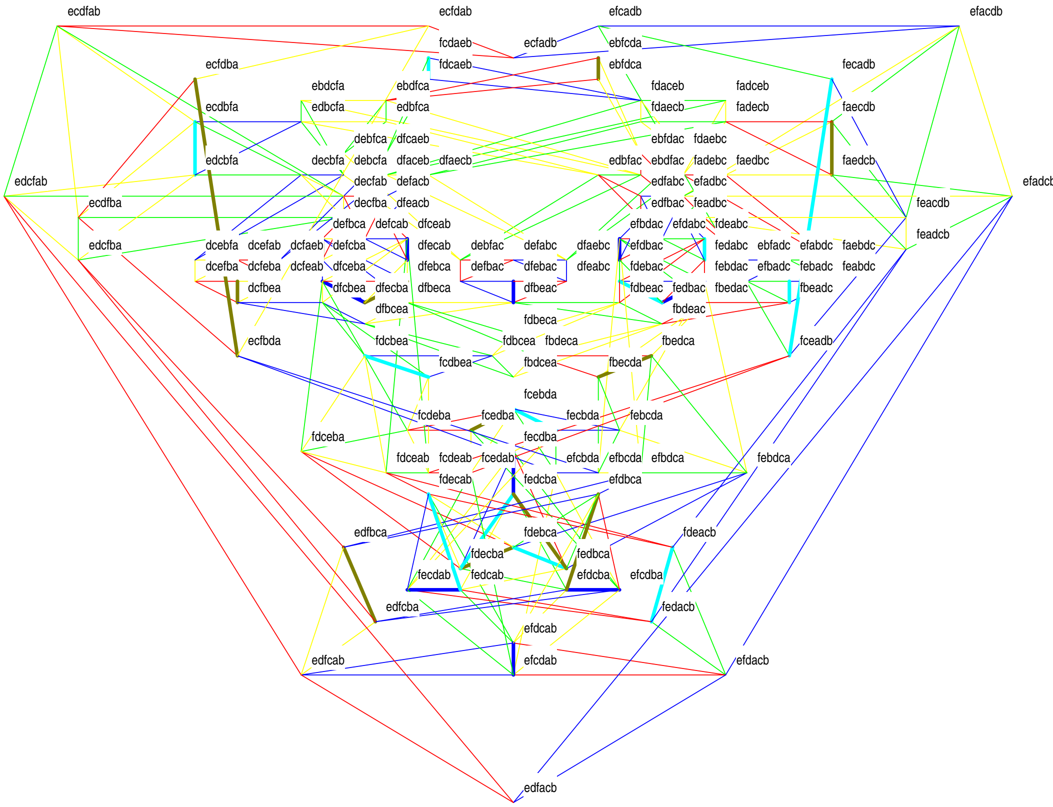


Figure 6: The great class *fedcba*: $\lambda(dcebfa) = abcdef$.

– If $\alpha \xrightarrow{3a} \beta$, then α replaces and disloges another small letter β, which climbs up in the staircase in its stead.

– If $\alpha \xrightarrow{3b} R$, then α becomes a neutral letter and stops.

– If $\alpha \xrightarrow{3c} R$, then α is inserted in a row where β was a neutral letter. α stays small, β becomes a great letter.

We now have a clear picture of the insertion algorithm: small letters bubble up in the staircase, get replaced by greater small letters, until they reach the row they index and become neutral letters, or finally disloge a neutral letter that becomes a great letter. One interesting point is that a small letter may become neutral or great through subsequent insertions, but a neutral letter can only stay neutral or become a great letter, and a great letter stays a great letter.

As an example, let us take a closer look at the words of the great class $Gr(6)$. Since *a*, *b*, *c* are small letters, by inspection of partial insertion staircases, we can determine what prefix a word in $Gr(6)$ can not have.

For instance, no word of $Gr(6)$ can start with *a*, *b*, or *c*. No word may start with *fab*, *fbc*, *fac* either. As a more general rule, a prefix may lead to a word of $Gr(6)$ only if it holds more great letters (*d*, *e*, *f*) than small letters (*a*, *b*, *c*). We will give in Section 4.2 a more detailed account of this fact.

We will now prove that the great class is the largest class of the Chinese monoid. In fact, it contains all the other subclasses in a very precise way.

Definition 4.4 *Let u, v, w be three words. An elementary rewriting between u, v, w is an elementary congruence of the Chinese relation: there exists three letters a, b, c, two words x, y such that $\{u, v, w\} = \{x.cba.y, x.cab.y, x.bca.y\}$. An* embedding *of a Chinese class C_1 into another Chinese class C_2 is an injection i from C_1 into C_2 that preserves the congruence graph. Specifically, for any elementary rewriting between words u, v, w of C_1, there exists an elementary rewriting between $i(u)$, $i(v)$, $i(w)$. The embedding i is* strict *if both elementary rewritings occur at the same position.*

Here is the outline of the proof: the following algorithm yields an embedding of every Chinese class which is not the great class into a greater class. Therefore, each class belongs to a sequence of increasing classes that ends necessarily at the great class.

Algorithm 4.5 (Class Embedding)
Let σ be a full standard staircase. The algorithm finds an embedding of $C(\sigma)$ into another Chinese class. Find β such that $\sigma_{\beta\rho(\beta)}$ be the right-most, non exposed entry. Let γ be the successor of β.

- if β does not exist, take the identity for an embedding (embed $C(\sigma)$ into itself.)
- if β does exist, let t_β^γ be the elementary tranposition which exchanges β and γ and leave other letters invariant. Take this transposition for an embedding.

Proposition 4.6 *This algorithm is correct, namely β does exist if and only if $C(\sigma)$ is not a great class, in which case t_β^γ is a strict embedding.*

Proof — The only full standard staircase without non exposed entries is the staircase of the great class. Otherwise β exists and can not be the greatest letter of the alphabet, therefore γ is well-defined as well.

In this case, let u, v and w be three words of $C(\sigma)$ that are congruent thanks to one elementary Chinese relation $cba \equiv cab \equiv bca$. If neither β nor γ occur in abc, the transposition t_β^γ does not affect the elementary Chinese relation at all. If only one of β and γ does occur in abc, the transposition t_β^γ does not change the relative order of abc, hence there still exists an elementary congruence between the images of cba, cab and bca. Assume now that β and γ occur in abc. Since β and γ are consecutive letters, there are only two cases to consider: $\beta < \gamma \le c$, so that $c\gamma\beta \equiv c\beta\gamma \equiv \gamma c\beta$ and $a \le \beta < \gamma$, so that $\gamma\beta a \equiv \gamma a\beta \equiv \beta\gamma a$. Since t_β^γ exchanges $c\gamma\beta$ and $c\beta\gamma$ on one hand, $\gamma\beta a$ and $\beta\gamma a$ on the other hand, the only problematic sequences are $\gamma c\beta$ and $\gamma a\beta$.

We contend that no word of $C(\sigma)$ contains such a sequence. To obtain the entry $\sigma_{\beta\rho(\beta)}$, the insertion algorithm must yield an insertion sequence that ends in inserting $\rho(\beta)$ into the row of β. Consider the hook of γ at this point.

- There can be no 1 in a column indexed by α with $\alpha > \rho(\beta)$, since insertion sequences yield increasing sequence of letters and since the entire hook of γ lies below the row of β.
- There can be no 1 in a cell $\sigma_{\gamma\alpha}$ with $\alpha < \rho(b)$: then we would have $\rho(\gamma) < \rho(b)$, and γ would be a non exposed letter, contradicting the definition of β.

The hook of γ is empty at this point, which means that γ occur after β in every word of $C(\sigma)$. Every elementary congruence is still valid after the transposition: by transitivity, the images of the words of $C(\sigma)$ are congruent. □

Theorem 4.7 *For a given n, all classes have an order less or equal to the cardinality of $Gr(n)$. In fact, for any Chinese class, there exists a permutation of A which is a strict embedding of this class into $Gr(n)$.*

Proof — Starting with the chinese class $C(\sigma_0)$, apply Algorithm 4.5 iteratively to define a sequence of larger and larger Chinese classes $C(\sigma_i)$. For $i \leq j$, the permutation we obtain is a strict embedding of $C(\sigma_i)$ into $C(\sigma_j)$.

Choose a word u in $C(\sigma_0)$. Let u_i be its iterated image in $C(\sigma_i)$. If $C(\sigma_i) \neq Gr(n)$, the algorithm supplies a transposition t_β^γ which puts β and γ in the right order, that is, the number of inversions of $u_{i+1} = t_\beta^\gamma(u)$ is strictly greater than the number of inversions of u_i. By exhaustion, there must eventually be a j such that $u_j \in Gr(n)$. □

Examples 4.8 Consider the class of *abcdefijhkg* of order 35. The algorithm embeds it successively into the classes of

bacdefijhkg (order 35),	*bcadefijhkg* (105),	*bcdaefijhkg* (175),
bcdeafijhkg (245),	*bcdefaijhkg* (315),	*bcdegaijhkf* (315),
bcdehaijgkf (315),	*bcdehiajgkf* (329),	*bcdehigjakf* (399),
bcdehigjfka (1 225),	*cbdehigjfka* (1 295),	*cdbehigjfka* (4 165),
cdebhigjfka (7 175),	*cdfbhigjeka* (7 175),	*cdgbhifjeka* (7 175),
cdghbifjeka (7 725),	*cdghfibjeka* (10 607),	*cdghfiejbka* (60 037),
dcghfiejbka (67 597),	*ecghfidjbka* (67 597),	*fcgheidjbka* (67 597),
fgcheidjbka (92 323),	*fgehcidjbka* (228 305),	

and finally *fgehdicjbka* (3 705 075) ! The embedding sends *abcdefghijk* to *kajbicdefgh*.

As a less formidable example, Figure 7 shows how various classes embed into $Gr(6)$. Compare Figure 6 and Figure 2.

4.2 Dyck words

In this Section, we use a special convention: a great letter designates a great letter of $Gr(A)$, that is, a letter in the upper half of A, and a small letter designates a small letter of $Gr(A)$, that is, a letter in the lower half of A.

We begin by studying $Gr(n)$ where $n = 2p$ is an even number. This partition in small letters/great letters and the structure of $Gr(n)$ are strongly related to the structure of Dyck words.

Definition 4.9 *Let $D = \{x, \bar{x}\}$ be the alphabet over two letters x and $\bar{x}$.*

- *The* height *of a word w over D is $h(w) = |w|_x - |w|_{\bar{x}}$.*
- *A word $w \in D^*$ is* a Dyck word *if every prefix word u of w verifies $h(u) \geq 0$ and if $h(w) = 0$.*
- *A word w is* a proper Dyck word *if every proper prefix word u of w verifies $h(u) > 0$ and if $h(w) = 0$.*
- *If w is a Dyck word, and u a prefix of w, we say that u is* a return to zero *of w if $h(u) = 0$ (Therefore, a proper Dyck word is a Dyck word with just one return to zero.)*

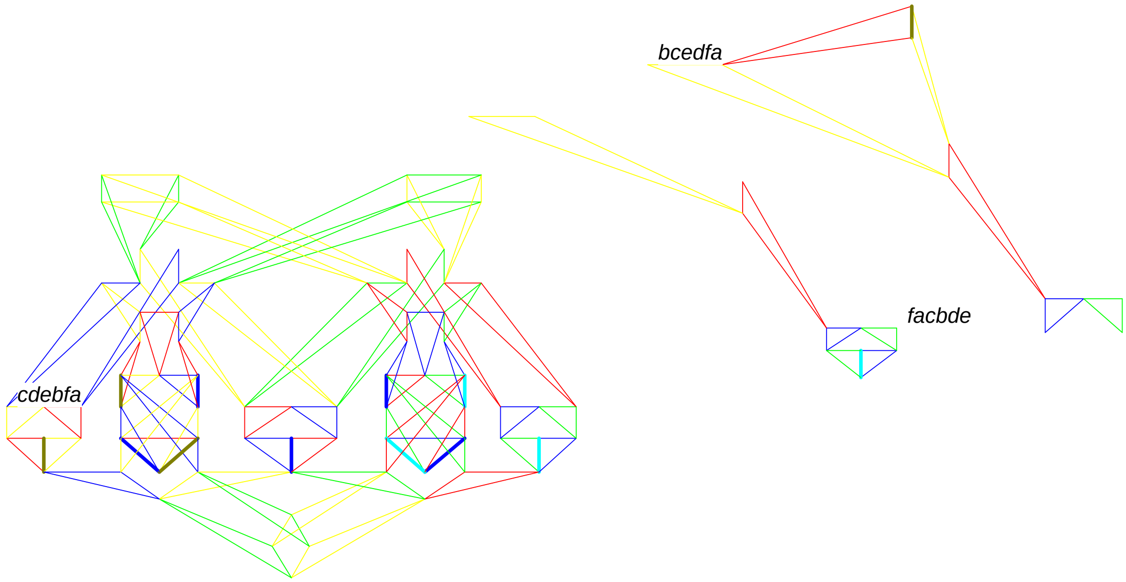


Figure 7: The transposition of *c* and *d* embeds the class of *cdebfa* (order 75) into the class $Gr(6)$ (order 135). Note that the embeddings of *facbde* and *bcedfa* do not preserve Schützenberger equivalence.

Definition 4.10 *Let A be an alphabet. We denote by π the monoid morphism defined by:*

$$\begin{array}{cccc} \pi: & A^* & \longrightarrow & D^* \\ & \alpha & \longmapsto & \begin{cases} x \text{ if } \alpha \text{ is a great letter of } Gr(A) \\ \bar{x} \text{ if } \alpha \text{ is a small letter of } Gr(A). \end{cases} \end{array}$$

Theorem 4.11 *Fix an alphabet A over $2p$ letters. Denote by c and c' the median letters of A; model $Gr(2p-2)$ over $A \setminus \{c, c'\}$. Then the great class $Gr(2p)$ is characterized as follows:*

- *if w belongs to $Gr(2p)$, if w' is obtained by deleting c and c' from w, then w' does belong to $Gr(2p-2)$.*
- *if w' belongs to $Gr(2p-2)$, if w is obtained by inserting c and c' in such a way that the image $\pi(w)$ is a Dyck word, then w does belong to $Gr(2p)$.*
- *if w' belongs to $Gr(2p-2)$, if w is obtained by inserting c and c' in such a way that the image $\pi(w)$ is not a Dyck word, then w does not belong to $Gr(2p)$.*

For example, *faeb* $\in Gr(4)$ and $\pi($*faeb*$) = x\bar{x}x\bar{x} \in D_4$. Insert *c* and *d* to obtain *fadebc* whose image by π is $x\bar{x}xx\bar{x}\bar{x}$ which is a Dyck word, hence *fadebc* belongs to $Gr(6)$. Insert *c* and *d* to obtain *facebd* whose image by π is $x\bar{x}\bar{x}x\bar{x}x$ which is not a Dyck word, and correspondingly, *facebd* does not belong to $Gr(6)$.

Proof — We write $w = a_1 \ldots a_j c a_{j+1} \ldots a_k c' a_{k+1} \ldots a_{2p-2}$, $w' = a_1 \ldots a_{2p-2}$.[6]

[6] Observe that we do not state which of $c < c'$ and $c > c'$ is true.

We need to compare what happens during the construction of the insertion staircase of w and during the construction of the insertion staircase of w'. Clearly, before the insertion of c, both staircases are exactly the same.

Lemma 4.12 *Let σ_i (resp. σ'_i) denote the staircase obtained by the partial insertion of w (resp. w') up to letter a_i. The left part of the staircase A_i (resp. A'_i) is the set of columns of σ_i (resp. σ'_i) indexed by letters greater than c and c', including the columns indexed by c and by c'. The right part of the staircase B_i (resp. B'_i) is the set of remaining columns, namely the set of columns indexed by letters smaller than c and c', not including the columns indexed by c and by c'.*

Then $B_i = B'_i$ and the 1 of A_i and A'_i occur on the same hooks.

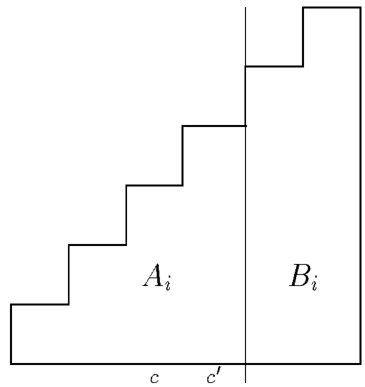


Figure 8: The definition of A_i and B_i.

The consequence of this lemma is that the lower 1 in A_i and A'_i are in the same row.

Start with the staircase corresponding to w', that is the staircase of the $Gr(2p-2)$ class. We have to check where we can insert c and c' in a word of $Gr(2p-2)$ to obtain a word of $Gr(2p)$. Observe that the height of a prefix of a Dyck word, as the difference between the number of great letters and small letters not yet inserted, is obviously the number of entries remaining in the left part of the staircase B'_i. Let $i = \inf\{i > j \mid \quad h(a_1 \dots a_i) = 0\}$.

We now have a precise picture of the sequence of staircases σ'_i: the height of the Dyck word corresponding to w' denotes the number of small letters still to be inserted in the staircase before the left part B'_i is left empty. There are several cases to consider:

- $c > c'$ and $k > i$. The entry corresponding to c is on the diagonal of the staircase. The same happens when c' is inserted except that, the first time the Dyck word reaches the bottom level after c' is inserted, the entry corresponding to c' moves up in the staircase until it reaches the entry corresponding to c. So the staircase σ is the same as the staircase σ' except for a 1 in $\sigma_{cc'}$. In this case, w belongs to $Gr(2p)$ and accordingly, its image $\pi(w)$ is a Dyck word.

- $c > c'$ and $k < i$. Each time the entry corresponding to c' moves up in the staircase and reaches c, c' stops and c starts climbing. Thus c' always remains under c. In this case, w belongs to $Gr(2p)$ and accordingly, its image $\pi(w)$ is a Dyck word.

- $c < c'$ and $k > i$. c ends up on the diagonal of σ, so σ cannot be the staircase of the great class. In this case, w does not belong to $Gr(2p)$ and accordingly, its image $\pi(w)$ is not a Dyck word.

- $c < c'$ and $k < i$. When c' is inserted, it does not necessarily climb higher than c (this depends on the left part A_i), but the insertion of a small letter forces c' to climb until finally c' goes above c; afterwards, c remains under c'. In this case, w belongs to $Gr(2p)$ and accordingly, its image $\pi(w)$ is a Dyck word. □

We now describe a procedure to compute the order of the great class.

Definition 4.13 *A* Dyck word with history *is a word w over the alphabet $\{x, \bar{x}\} \times \mathbb{N}$ that verifies:*

- *If $|w| = 2p$, w is a permutation of the letters $x_1, \dots x_p, \bar{x}_1, \dots, \bar{x}_p$.*
- *The monoid morphism $\nu : x_i \mapsto x$, $\bar{x}_i \mapsto \bar{x}$, sends w to a Dyck word.*
- *The word w' obtained by deleting x_p and $\bar{x}_p$ from w is a Dyck word with history (the empty word is a Dyck word with history.)*

In other words, a Dyck word with history embodies a Dyck word and a way to build it through successive Dyck words.

Corollary 4.14 *Fix an alphabet A over $2p$ letters. Let ζ be the monoid isomorphism between A^* and $D \times \{1, \dots, p\}$ that sends the small letters of A $a, b, \dots$ to $\bar{x}_1, \bar{x}_2, \dots$ and the great letters of A $z, y, \dots$ to $x_p, x_{p-1}, \dots$.*

Then ζ is a bijection between $Gr(2p)$ and the Dyck words with history of length $2p$.

Actually, we can count the words of $Gr(2p)$ directly by associating a weight to Dyck words.

Definition 4.15 *A Dyck word of length $2p$, $p > 0$, can be reduced to a Dyck word of length $2p-2$ in a variety of ways by deleting one x and one $\bar{x}$. Each of these ways will be called a Dyck reduction. We define* the weight *of a Dyck word inductively:*

- *$Weight(x\bar{x}) = 1$.*
- *The weight of a Dyck word of length $2p$ is the sum of all the weights of Dyck words obtained by all possible Dyck reductions.*

Example 4.16 Since $xx\bar{x}\bar{x}$ reduces as $\not{x}x\not{\bar{x}}\bar{x}$, $\not{x}x\bar{x}\not{\bar{x}}$, $x\not{x}\not{\bar{x}}\bar{x}$, and $x\not{x}\bar{x}\not{\bar{x}}$, $Weight(xx\bar{x}\bar{x}) = 4$. On the other hand, $x\bar{x}x\bar{x}$ reduces as $\not{x}\not{\bar{x}}x\bar{x}$, $x\not{\bar{x}}\not{x}\bar{x}$, and $x\bar{x}\not{x}\not{\bar{x}}$, so $Weight(x\bar{x}x\bar{x}) = 3$. Similarly the reader can check that $Weight(xxx\bar{x}\bar{x}\bar{x}) = 36$.

We now state our main result concerning the cardinality of the great class.

Theorem 4.17 *For every Dyck word w of length n, $Weight(w)$ verifies:*

$$Weight(w) = |\{u \in Gr(n),\ \pi(u) = w\}|.$$

Proof — By definition, $Weight(w) = |\nu^{-1}(w)|$. □

Corollary 4.18 *The sum of the weights of all Dyck words of n letters is equal to the cardinality of $Gr(n)$.*

Let us now assume that n is odd, $n = 2p + 1$. A similar argument yields the following theorem.

Theorem 4.19 *Fix an alphabet A over $2p + 1$ letters. Denote by c the median letter of A; model $Gr(2p)$ over $A \setminus \{c\}$. Then the great class $Gr(2p+1)$ is characterized as follows:*

- *if w belongs to $Gr(2p+1)$, if w' is obtained by deleting c from w, then w' does belong to $Gr(2p)$.*
- *if w' belongs to $Gr(2p)$, if w is obtained by inserting c in w, then w does belong to $Gr(2p + 1)$.*

Corollary 4.20 *If n is an odd integer. Then $|Gr(n)| = (n)\,|Gr(n-1)|$.*

5 Generalization to the other classes

The idea of inserting letters can be generalized to all the classes in the standard case, as this Section shows.

Definition 5.1 *Let M be the alphabet over three letters $x, \bar{x}$ and t. We consider the monoid morphism μ from M^* to D^* defined by $\mu(x) = x$, $\mu(\bar{x}) = \bar{x}$, $\mu(t) = \epsilon$. A word w of M^* is called* a Motzkin word *if $\mu(w)$ is a Dyck word. The* type *of a Motzkin word w is $(|w|_x, |w|_t)$. Thanks to μ, we obtain a natural notion of height of a Motzkin word, proper Motzkin words, and returns to zero.*

Definition 5.2 *Let C be a congruence class of the Chinese monoid. The type of C is (i, j) where i is the number of pairs of great and small letters and j is the number of neutral letters.*

We define like we did in the previous section the morphism that sends every word of a Chinese class to a Motzkin word of the same type.

Definition 5.3 *Let A be an alphabet, C a Chinese class over A. The monoid morphism π is defined by:*

$$\begin{array}{cccl} \pi: & A^* & \longrightarrow & M^* \\ & \alpha & \longmapsto & \begin{cases} x \text{ if } \alpha \text{ is a small letter of } C \\ \bar{x} \text{ if } \alpha \text{ is a great letter of } C \\ t \text{ if } \alpha \text{ is a neutral letter of } C. \end{cases} \end{array}$$

Definition 5.4 *Let σ be a standard staircase. The first non null entry $\sigma_{\beta\alpha}$ in the row-reading order is called* the central entry of the staircase*, that is, for any $\beta' < \beta$, any α', $\sigma_{\beta'\alpha'} = 0$, for any $\alpha' < \alpha$, $\sigma_{\beta\alpha'} = 0$, and $\sigma_{\beta\alpha} > 0$. The letters α, β corresponding to this entry are called* the central letters of the staircase *(note that we may have $\alpha = \beta$.) All entries located in the lower-left part of the staircase relative to the central entry are called* external entries. *The corresponding letters are called* the external letters of the staircase. *The staircase $D\sigma$ obtained by setting $\sigma_{\beta\alpha}$ to zero is called* the derivative staircase of σ*, and the corresponding class DC* the derivative class of C.

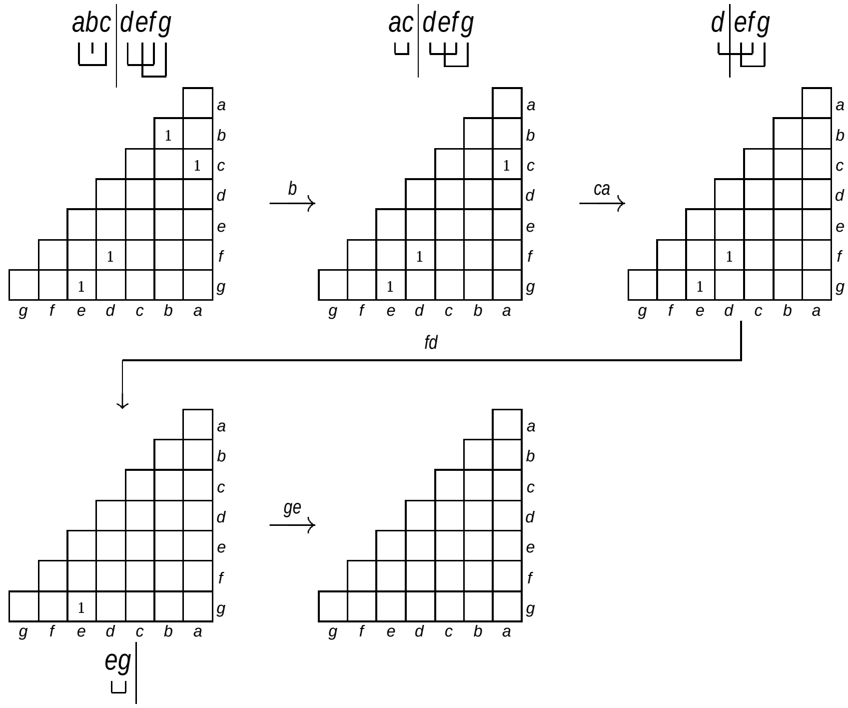


Figure 9: The successive derivations of *bcafdge*.

5.1 Two reduction theorems

Theorems 4.19 and 4.11 generalize to two reduction theorems $\mid$ and $\sqcup$, depending whether there is one neutral central letter or there is a pair of two central letters.

Theorem 5.5 ($\mid$ — one central letter.) *Let σ be a standard staircase with central entry σ_c. Let e_j be the external letters of σ, s_i the letters smaller than q, and let $g_i = \rho(s_i)$ be the associated letters.*

Then all letters s_i are small letters, and the sets $\{s_i\}$, $\{g_i\}$, $\{e_j\}$ and $\{c\}$ form a partition of the alphabet A.

A word $w = a_1 \dots a_k c a_{k+1} \dots a_{n-1}$ belongs to $C(\sigma)$ if and only if

- *$w' = a_1 \dots a_{n-1}$ belongs to $C(D\sigma)$,*
- *If M is the Motzkin word associated with w, if p is the position of the first return to 0 of M after the letter t associated with c, all the letters e_j occur after position p in w.*

Proof — Notice that these two inequalities hold: $s_i < c < g_i$ and $c < e_j$, so that in particular, c is smaller than the small letters of the set $\{e_j\}$. The direct part can be proved by using the same argument as in the case of the great class: the final place of the 1 corresponding to s_i and s_i are in the right part of the staircase. The trick is the following: thus, in the proper Dyck word which contains c, the height denotes the number of letters smaller than c that have to be inserted before the 1 associated with c reaches its final position in σ. Thus if a letter associated to one of the e_j is inserted the 1 cannot reach its place since c is smaller than the small letters of the set $\{e_j\}$. □

With a similar argument we prove the following theorem:

Theorem 5.6 **(⊔ — two central letters.)** *Let σ be a standard staircase central entry $\sigma_{c_1 c_2}$. Let e_j be the external letters of σ, s_i the letters smaller than c_2 and $g_i = \rho(s_i)$ be the associated letters.*

Then all letters s_i are small letters, and the sets $\{s_i\}$, $\{g_i\}$, $\{e_j\}$ and $\{c_1, c_2\}$ form a partition of the alphabet A.

A word $w = a_1 \dots a_k c a_{k+1} \dots a'_k c' a_{k'+1} \dots a_{n-2}$ belongs to $C(\sigma)$ if and only if

- *$w' = a_1 \dots a_{n-2}$ belongs to $C(D\sigma)$.*
- *$\{c, c'\} = \{c_1, c_2\}$.*
- *If M is the Motzkin word associated with w, if p is the position of the first return to 0 of M after the letter x or $\bar{x}$ associated with c', all the letters e_j occur after position p in w.*

5.2 Robinson-Schensted correspondence

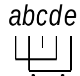
We still have to find a way of inserting t and pairs of $x, \bar{x}$ to obtain a Motzkin word, while preserving the structure inferred from the reduction theorems.

Definition 5.7 *A* Motzkin word with history of type (n, p) *is a word w over the alphabet $M \times \mathbb{N}$ that verifies:*

- *The word w is a permutation of the letters $x_1, \dots, x_n, \bar{x}_1, \dots, \bar{x}_n, t_1, \dots, t_p$.*
- *The monoid morphism ν: $x_i \mapsto x$, $\bar{x}_i \mapsto \bar{x}$, $t_p \mapsto t$ sends w to a Motzkin word.*
- *The word w' obtained by deleting x_n and $\bar{x}_n$ from w is a Motzkin word with history of type $(n-1, p)$.*
- *The word w' obtained by deleting t_p from w is a Motzkin word with history of type $(n, p-1)$.*
- *The empty word is a Motzkin word with history of type $(0, 0)$.*

Similarly to the case of the great class, the row normal form of a class yields an ordering of the pairs of great/small letters. For a class C of a given type, there is a one-to-one correspondence between the letters of a well ordered Motzkin word of the same type and the letters of the alphabet. By applying the converse of this correspondence, each Motzkin word with history of the proper type yields a word. We will now examine under what condition this word belongs to C.

Example 5.8 Consider the class C of *cdaeb*, of link representation *abcde*.



The Motzkin word with history $x_1 x_2 t_1 \bar{x}_2 \bar{x}_1$ has the right type, but it is associated with the word *edcab*, which is not in C.

Definition 5.9 *Let C be a Chinese class, w a Motzkin word with history of the same type. The word w corresponds to a word of C. Define the derivative word Dw of w by deleting the letter(s) corresponding to the central entry of C in w.*

*The word w is C-*admissible *if and only if*

- *the word Dw is a Motzkin word with history.*
- *The word Dw is DC-admissible,*
- *If there is one central letter, it corresponds to t_1.*
- *The letters of w corresponding to the external letters of C occur after the first return to 0 of w after the letter(s) associated with the central letter(s) of C.*

Just remark that t_1 is a-admissible and $x_1\bar{x}_1$ is ba-admissible (we denote the class by its normal form).

Using Theorem | and Theorem ⊔ it is easy to prove the following.

Theorem 5.10 *Let C be a Chinese class of type (n,p) , let t be its row normal form. Let ζ be the monoid isomorphism between A^* and $\{x,\bar{x}\} \times \{1,\dots n\} \ \cup \ \{t\} \times \{1,\dots,p\}$ that sends the great letters of A to $x_1,\dots x_n$ according to their respective positions in t, the neutral letters of A to $t_1,\dots,t_p$ according to their respective positions in t, and the small letters of A to $\bar{x}_1,\dots \bar{x}_n$ according to the respective positions of the corresponding great letters by ρ in t.*

Then ζ is a bijection between the words of C and the C-admissible Motzkin words with history.

We want to stress the following technical point: the return to zero of the Motzkin word associated to w' cannot be read on w; take a look at this example: in the class of *edfbgcha* the word $x_1\bar{x}_1x_3x_4\bar{x}_3x_2\bar{x}_4\bar{x}_2$ is admissible, but $x_1x_3\bar{x}_3\bar{x}_4x_4x_2\bar{x}_2\bar{x}_1$ is not since the deletion of x_4 and $\bar{x}_4$ moves the position of the return to 0 to the end of the word.

5.3 Counting patterns

We now want to count the number of classes of a given order in the standard case. Since the number of distinct classes is quite formidable, we will need to merge cases by introducing a notion of isomorphism between classes, that is, by identifying common patterns. We begin by considering the case of order 1 classes, where a simplified version of the general idea applies: we are going to split a staircase into primitive staircases that do not interfere with one another. Staircases of order one play a central role in the decomposition.

Lemma 5.11 *Let σ be the staircase of a class of order one. Then σ does not contain two 1 such that one lies to the north-west of the other.*

Proof — Remember the converse algorithm (Algorithm 3.6.) Since the class holds just one element, only one rewriting must apply at each step, hence the set of deletable elements must always hold just one element. □

We may transpose this lemma to the corresponding word:

Proposition 5.12 *A word w belongs to a class of order 1 if and only if both the sequence of letters at odd positions in w and the sequence of letters at even positions are increasing sequences.*

Example 5.13 Take $w = \textsf{abcdgehf}$. Since both *acgh* and *bdef* are increasing sequences, the class of w has order one.

Corollary 5.14 *The number of classes of order one with n distinct letters is:* $\binom{n}{\lfloor n/2 \rfloor}$.

Consider a Chinese class $C(\sigma)$ and examine carefully the positions a given letter can occupy. In classes of order 1, all letters have a fixed position. In some classes, only some letters move. In other classes, all letters move, but most of them only occupy a limited range of positions.

Definition 5.15 *Consider two staircases σ and σ' over the alphabets A and A'. We say that σ and σ' hold the same pattern if there exist an increasing injection i from A to A' and an increasing injection j from A' to A such that $\sigma = \sigma' \circ (i, i)$ and $\sigma' = \sigma \circ (j, j)$.*

Proposition 5.16 *If two staircases hold the same pattern, they have isomorphic support. Formally a pattern is a staircase up to isomorphism of its support set.*

Proof — Let S be the support of σ, that is the set of letters α such that there exist a non null $\sigma_{\alpha\beta}$ or $\sigma_{\beta\alpha}$, let S' be the support of σ'. Then i and j define an isomorphism of $(S, <)$ and $(S', <)$. □

Definition 5.17 *Let σ be a staircase over alphabet A, let B be a subset of A. The* subpattern P of σ over B *is the staircase over B obtained by taking all the cells of σ with both indices in B.*

This does not make much sense in the general case. An extractible pattern *is a subpattern P over B such that all other cells of σ with one index in B have empty contents.*

An extractible pattern P is a right subpattern *if and only if all cells to the north-est of P vanish. An extractible pattern is a* left subpattern *if and only if all cells to the south-west of P vanish.*

A pattern is a primitive pattern *if it does not contain any non trivial extractible subpattern.*

Example 5.18 Consider a standard word w such that $w = uv$, with all letters of u less than all letters of v. Then the subpattern U over the letters of u is extractible, the subpattern V over the letters of v is extractible. The class $C(w)$ is a product class. Indeed, $C(w) = C(U).C(V)$.

Obviously a given staircase decomposes into a collection of primitive extractible subpatterns in a unique way.

Definition 5.19 *Consider a pattern P.* The left part of P $l(P)$ *is the set of small letters such that all preceding letters in lexicographic order that belong to P are small.* The height of P at the left $hl(P)$ *is the order of $l(P)$.* The right part of P $r(P)$ *is the set of great* *letters such that all succeeding letters in lexicographic order that belong to P are great.* The height of P at the right $hr(P)$ *is the order of* $r(P)$.

For example, the pattern *bcdea* of link representation *abcde* verifies

$$hl(bcdea) = hr(bcdea) = 1.$$

The pattern *cbdae* of link representation *abcde* verifies

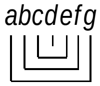
$$hl(cbdae) = 1, \; hr(cbdae) = 0.$$

Proposition 5.20 *Let P be a pattern over p letters. The number of staircases σ with n letters ($n \geq p$) that have P as a left pattern, and such that the rest of σ is composed of primitive patterns of order one is*

$$\binom{hr(P)+n-p}{\lfloor \frac{n-p}{2} \rfloor}.$$

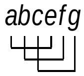
Proof — Consider the following transformation of the pattern P. The pattern ends in $hr(P)$ great letters. We build a new pattern P' by associating these letters with suitable small letters such that $C(P')$ has order one. For example, consider the pattern *decfbga*.

abcdefg.



The right part of P is $r(P) =$ *efg*. Associate *a* with *e*, *b* with *f*, *g* with *c*, remove *d* to obtain the new pattern $P' =$ *eafbgc*.

abcefg.



Then we apply the same transformation of this subpattern to all the staircases σ as defined in the proposition. The classes we obtain are obvious classes of order one on $2hr(P) + n - p$ letters, with the supplementary condition that the first $hr(P)$ smallest letters must occur at the first even positions in the canonical word. To have classes of order one, we have to choose which letters will occur at even positions in the row normal form; this yields $hr(P) + \lfloor \frac{n-p}{2} \rfloor$ possibilities. Since the first $hr(P)$ even letters are fixed, there still remains to choose $\lfloor \frac{n-p}{2} \rfloor$ among the $hr(P) + n - p$ final letters. This yields the desired result. □

Proposition 5.21 *Let P and Q be two patterns with respectively p and q letters. The number of classes with n letters ($n \geq p+q$) that have P as a left pattern and Q as a right pattern and whose other primitive patterns are of order one is:*

- $\binom{hr(P)+hl(Q)+n-p-q}{hl(Q)+\frac{n-p-q}{2}}$ *if $n-p-q$ is even,*
- $\binom{hr(P)+hl(Q)+n-p-q}{\lfloor \frac{n-p-q}{2} \rfloor}$ *if $n-p-q$ is odd.*

Definition 5.22 *We take the following ordering on ones of a staircase. A 1 is smaller than another 1 if and only if the first one has its great letter lesser than the great letter of the second one and greater than the small letter of the second one.*

The order can be seen in a very simple way when using the link representation.

Example 5.23 For the staircase $cbda$, we have $cb < da$. For the staircase $cadb$, we have $ca < db$.
We call *pyramidal pattern* a pattern in which the 1 are not in the diagonal and can be ordered in a chain such that every element of the chain is smaller than the previous element.

Definition 5.24 *Let P be a staircase composed with a pattern, a left part which is a class of order one and a right part which is a class of order one. There is an interference between the right and the left part if and only if there is a small letter of the right part that is smaller (in the lexicographic order) than a great letter of the left end. The number of interferences is the greatest number of letters g in the left part of σ which can be associated to different letters s in the right part of σ such that $g > s$.*

Theorem 5.25 *Let P be a pattern that is not a pyramidal pattern. Then the number of staircases σ with q letters that have P as a pattern and such that all the other patterns of σ that are disjoint of P are of order one is the following sum:*

$$\sum_{j=0}^{q-n} \binom{hl(P)+j}{\lfloor \frac{j}{2} \rfloor} \binom{hr(P)+q-n-j}{\lfloor \frac{q-n-j}{2} \rfloor}.$$

Proof — In the previous sum, j denotes the number of letters at the left of the pattern. The proof is clear since that if P is not a pyramidal pattern, the letters involved in $hr(P)$ and $hl(P)$ have no interferences. In particular, j being chosen, the choice of the letters involved in the left and the right part are totally independant. This then gives the wanted product. □

We now have to work on the number i of interferences between the right and the left of the pattern in the cases of the pyramidal patterns.

Theorem 5.26 *Let P be a pyramidal pattern. The number of staircases σ with q letters that have P as a pattern and such that all the other patterns of σ that are disjoint of P are of order one is:*

$$\sum_{i=0}^{\lfloor \frac{q-n}{2} \rfloor} \sum_{j=0}^{q-n-2i} \binom{hl(P)+i+j}{\lfloor \frac{j}{2} \rfloor} \binom{hr(P)+i+q-n-j}{\lfloor \frac{q-n-j}{2} \rfloor}.$$

Proof — If there are i interferences, we can transform P by adding i to $hr(P)$ and $hl(P)$, and count like in the former theorem supposing there are no interferences. □

Note 5.27 With all these formulas, one can compute the number of classes with n letters that have a given order. To do this, one has just to know all the extracted patterns that exist.

6 Conjugacy classes

This Section investigates into some properties of the Chinese monoid that are not related with anything else in this paper.

Definition 6.1 *The conjugacy relation $\sim$ on the Chinese monoid is the transitive closure of the relation R defined by:*

$$c \; R \; c' \Leftrightarrow \exists u, v \; \in \; Ch(A)\,, \; c = uv, \; c' = vu,$$

for every c, $c' \in Ch(A)$.

Note 6.2 Consider the following two operators on words:

- Operator E: take the chinese congruence class of an element.
- Operator Rot: take the last letter of a word and put it in first position.

The conjugacy class of w is the set of all words that can be obtained by a finite number of applications of E or Rot.

The following result gives the structure of the Chinese conjugacy classes. It is an obvious consequence of the more precise Theorem 6.5, to follow.

Theorem 6.3 *The conjugacy class of a word for the Chinese congruence is its evaluation class, that is to say, the set of all words that have the same numbers of $a, b, \ldots, z$ as w.*

Definition 6.4 *We define yet another equivalence relation on words:*
$a_1 a_2 \ldots a_k = w \approxeq w' = a'_1 a'_2 \ldots a'_k$ *if and only if:*

- $w \equiv w'$ *or*
- $a_{k-1} a_k a_1 \equiv a'_{k-1} a'_k a'_1$ *or*
- $a_k a_1 a_2 \equiv a'_k a'_1 a'_2$,

That is to say: we allow elementary Chinese congruences to wrap around (picture words written on a circle with a pointed position to mark the beginning of the word). Figure 10 shows the wrapping Chinese classes for $n = 4$.

The wrapping Chinese congruence $\approxeq$ holds the key to the conjugacy relation $\sim$, since it is obviously coarser than the Chinese congruence $\equiv$ but finer than the conjugacy relation $\sim$.

For instance, *badc* $\approxeq$ *cadb* $\approxeq$ *bacd* since *dcb* $\equiv$ *dbc* $\equiv$ *cdb*.

Theorem 6.5 *Consider the standard case for an alphabet A of n letters.*

1) *If n is odd, there is just one wrapping Chinese class.*
2) *If n is even, there are exactly two wrapping Chinese classes. Moreover Rot swaps these classes.*

Proof — First notice that the wrapping Chinese class of a word is included in its conjugacy class. We define the following algorithm:

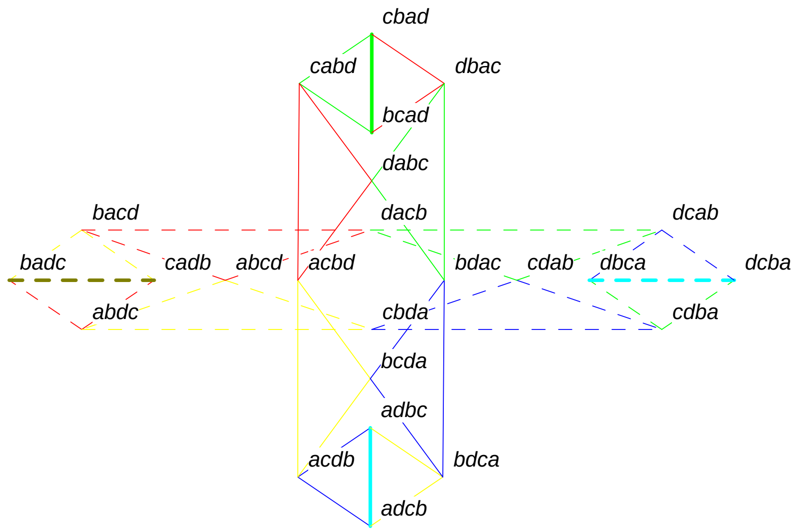


Figure 10: the wrapping Chinese congruence over 4 letters

Algorithm 6.6 (Standardization for the wrapping Chinese congruence) Let w be a standard word, of length $|w| > 2$, a the smallest letter of w, z the greatest letter of w.

- Step 1: move the letter a to the position just to the right of z using the elementary congruences of $\approx$: $cba \approx cab$ and $bca \approx cab$ (a slides to the left.)
- Step 2: move the block za to the beginning of w using the elementary congruence template $bca \equiv cab$ as in $y(za) \approx (za)y$.[7] (za slides to the left.)

Applying this algorithm yields a word $w' = zau$. We then apply the algorithm recursively to u which is 2 letters shorter. The main point is that a and z are the two extreme letters of the alphabet, hence every other letter can cross the block za without problems. if n is even, iterated application of Algorithm 6.6 to w clearly yields

- either $w \approx zayb \dots \alpha\beta$,
- or $w \approx zayb \dots \beta\alpha$,

with α and β the median letters of A. If n is odd, iterated application of Algorithm 6.6 yields $zayb \dots \alpha$ with α the median letter of A.

Let us wrap the case 'n even' up. We have shown that there are at most two classes. There remains to prove that there are exactly two classes. To achieve this, since there are two distinct normal forms, we only have to show that if two words are congruent by $\approx$, Algorithm 6.6 yields the same result. It is enough to prove this result for an elementary $\approx$-congruence, which is immediate. Then the set of wrapping Chinese classes is stable by *Rot*, and it is easy to check that the image of a class by *Rot* cannot be itself. For instance, take the case $n = 6$. The class [*faebdc*] is sent to the class [*faebcd*] since *cfaebd* $\approx$ *faebcd* as *c* can cross every couple of letters. Conversely, we can prove that the class [*faebcd*] is sent to the class [*faebdc*]. A similar argument applies for any even n. □

[7] Since all other letters are greater than a and smaller than z.

Note 6.7 The *center* of the Chinese monoid is the set of words that commute with every other word in the insertion algorithm, As announced in [2], we can easily check that the center is:

$$(za)^*.$$

Note 6.8 In the case of the Plactic monoid, based on experimental results, we conjecture that the number of $\approx$-classes is exactly the number of letters in the standard case.

Conclusion

This paper mostly presents some powerful tools to deal with enumeration problems regarding the Chinese monoid. There are still lots of open questions. For instance, a closed formula for the order of the great class.

This study relies a lot on actual experiments. In order to understand the Chinese monoid, being able to obtain graph representations and manipulate them easily was of great help.

Accordingly, we developped a suit of programs that will be made available shortly on `ftp://ftp.ens.fr/pub/dmi/users/espie/chinese`. These programs are written in Icon, a very powerful high level language. Icon's advanced features—generators, polymorphic data-types, graphics functions—enabled us to develop simple experimental code, very easy to modify and adapt to our needs in a small fraction of the time we would have needed using a more conventional programming language.

We would like to thank the people from the University of Arizona Computer Science Department for their very fine work (Griswold, [13]). More information about Icon is available from `ftp://cs.arizona.edu/icon`.

References

[1] DATE E., JIMBO M., MIWA T., *Representations of $U_q(gl_n)$ at $q = 0$ and the Robinson-Schensted correspondance*, [in "Physics and mathematics of strings, Memorial volume of V. Knizhnik", L. Bring, D. Friedan, A.M. Polyakov, Eds.], World Scientific, 1990.

[2] DUCHAMP G., KROB D., *Plactic-growth-like monoids*, [in "Words, languages and combinatorics II", Kyoto, Japan, 25-28 August 1992, M. Ito, H. Jürgensen, Eds.], 124–142, World Scientific, 1994.

[3] GREENE C., *An extension of Schensted's theorem*, Adv. Maths., **14**, 254–265, 1974.

[4] KNUTH D.E., *Permutations, matrices and generalized Young tableaux*, Pacific J. Math., **34**, 709–727, 1970.

[5] KNUTH D.E., *The Art of computer programming, vol. 3, Sorting and Searching, Addison Wesley, 1976*

[6] LASCOUX A., SCHÜTZENBERGER M.P., *Le monoïde plaxique*, [in "Non-commutative structures in Algebra and Geometric Combinatorics", A. De Luca, Ed.], Quaderni de la ricerca scientifica, **109**, 129–156, Rome, 1981.

[7] Lascoux A., Schützenberger M.P., *Keys and standards bases*, [in "Invariant theory", D. Stanton, Ed.], Springer, 1990.

[8] Leclerc B., Thibon J.Y., *The Robinson-Schensted correspondance as the quantum straightening at* $q = 0$, LITP Technical Report 95/17, Paris, 1995.

[9] Macdonald I.G., *Symmetric functions and Hall polynomials*, Oxford, 1995.

[10] Robinson G.B. De, *On the representation theory of the symmetric group*, Amer. J. Math., **60**, 745-760, 1938.

[11] Schensted C., *Longest increasing and decreasing subsequences*, Canad. J. Math., **13**, 179–191, 1961.

[12] Schützenberger M.P., *La correspondance de Robinson-Schensted*, [in "Combinatoire et représentations du groupe symétrique", D. Foata, Ed.], Lect. Notes in Maths., **579**, 59–113, Springer, 1976.

[13] Griswold R.E., Griswold M.T., *The Icon Programming Language*, Prentice-Hall, second edition, 1990.