THE CHINESE MONOID

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

January 8, 1999

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. Entre autres, nous avons trouvé un plongement de chacune des classes d'équivalence dans la plus grande classe. Quant à la dernière partie de cet article, elle 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 relation scheme $cba \equiv bca \equiv cab$. An algorithm similar to Schensted's algorithm yields a characterization 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 largest one. Finally, the end of this paper is devoted to the study of conjugacy classes.

[*] `cassaign@iml.univ-mrs.fr` Julien Cassaigne, IML – Case 907 – 163, avenue de Luminy – 13288 Marseille Cedex – France

[†] `Marc.Espie@liafa.jussieu.fr` Marc Espie, LIAFA – Université Paris 7 – 2, Place Jussieu – 75251 Paris Cedex 05 – France

[‡] `Florent.Hivert@ens.fr` Florent Hivert, IGM – Université de Marne-la-Vallée – 2, rue de la Butte Verte – 93166 Noisy-le-Grand Cedex – France

[§] `Daniel.Krob@liafa.jussieu.fr` Daniel Krob, LIAFA

[¶] `Jean-Christophe.Novelli@liafa.jussieu.fr` Jean-Christophe Novelli, LIAFA

Introduction

This paper is devoted to the study of the so-called *Chinese monoid* which is the quotient of the free monoid over a totally ordered alphabet $(A, <)$ by the congruence generated by the Chinese relations that are defined by setting

$$\begin{aligned} aba \equiv baa,\ bba \equiv bab \quad & \text{for every } a < b\ , \\ cba \equiv bca \equiv cab \quad & \text{for every } a < b < c\ . \end{aligned} \tag{1}$$

In order to understand our motivations, we must retrace the story of the well known plactic monoid to which the Chinese monoid is strongly related.

We shall first go back to Schensted's seminal paper of 1961 (see [14]). In this paper, Schensted introduced an algorithm (now called Schensted's algorithm) for solving the elementary algorithmic problem of finding the length of the longest increasing subsequence of a sequence of integers.

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 (see [3]). It is also worth noting that Schensted's algorithm is the main ingredient of the Robinson-Schensted correspondence which is a fundamental bijection between words and pairs (P, Q) of Young tableaux of the same shape (see [13]).

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

$$\begin{aligned} aba \equiv baa,\ bba \equiv bab \quad & \text{for every } a < b\ , \\ acb \equiv cab,\ bca \equiv bac \quad & \text{for every } a < b < c \end{aligned} \tag{2}$$

(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 Schützenberger who decided to call *plactic monoid* the monoid defined by the relations (2) (see [8]). 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 an irreducible representation of $\mathfrak{S}_n$ of a tensor product of two irreducible representations of $\mathfrak{S}_n$) (see [15, 11]). Lascoux and Schützenberger then developed 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 [8, 9] for instance).

More recently Date, Jimbo and Miwa discovered a strong relationship between the usual Robinson-Schensted correspondence and the quantum group $U_q(gl_n)$ (see [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 $\{ a_1, \ldots, a_n \}^*$, with $a_1 < \cdots < a_n$. 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 (see [10] for details).

This last result shows that the plactic monoid plays in some sense the role of a maximal torus for $U_q(gl_n)$. Another generalization of the plactic monoid, the so-called hypoplactic monoid (see [6], [7] and [12]), can be interpreted in the same way. The hypoplactic monoid is the quotient of the plactic monoid by some additional quartic relations and one can show that it plays the role of the maximal torus for the quantization of $U_q(Gl_n)$ introduced by Dipper and Donkin.

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 (see [8]), but none was based on growth properties. The Hilbert series of the plactic monoid is indeed given by Schur-Littlewood formula. In other words,

$$\sum_{I \in \mathbb{N}^n} p_I \, a^I = \frac{1}{\prod_{a \in A} (1-a) \prod_{a<b \in A} (1-ab)} \,, \tag{3}$$

where p_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 (3). But only two of these three families are regular: the plactic monoid, and the so-called Chinese monoid. This Chinese monoid appeared moreover 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 and our paper constitutes the first combinatorial study devoted to this new monoid.

The main question remains in fact open. As we pointed out before, the importance of the plactic monoid can be really understood from the quantum point of view which was only discovered about 20 years later after its discovery ! There is certainly a high probability that the Chinese monoid is also the quantum maximal torus of some unknown quantum group. It may therefore be considered as an exciting challenge to identify this "Chinese" quantum group ...

Finally, let us present the structure of our paper. 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 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. Finally Section 5 is devoted to the study of the conjugacy relation : we prove that, as in the plactic case, conjugacy classes are identical to evaluation classes.

Acknowledgments

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*

$$cba \equiv cab \equiv bca \qquad \text{for every } a < b < c, \tag{4}$$

$$aba \equiv baa\,,\ \ bba \equiv bab \qquad \text{for every } a < b. \tag{5}$$

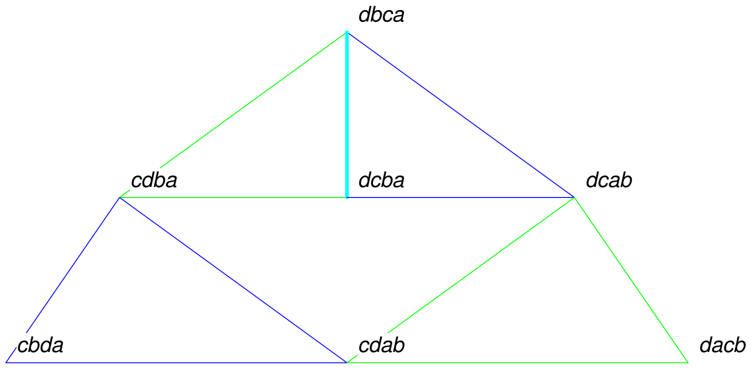
The Chinese monoid $CH(A,<)$ *is the quotient monoid of A^* by the Chinese congruence.*

Note 1.2 Obviously, (4) and (5) together are equivalent to

$$cba \equiv cab \equiv bca \qquad \text{for every } a \le b \le c. \tag{6}$$

Therefore we will almost exclusively use (6) 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 congruence relation. The thicker 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:

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

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

$$u \equiv_< v \quad \Leftrightarrow \quad \natural(u) \equiv_> \natural(v), \tag{7}$$

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 A^*. Choose $a \le b \le c$ as in (6). We have

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

for the Chinese congruence $\equiv_{>}$.

For an alphabet A over the ordered letters $a, b, \ldots, z$ consider the elementary morphism $\mathcal{I}: (A^*,<) \to (A^*,>)$ defined by $\mathcal{I}(a) = z, \mathcal{I}(b) = y, \ldots, \mathcal{I}(z) = a$. 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.

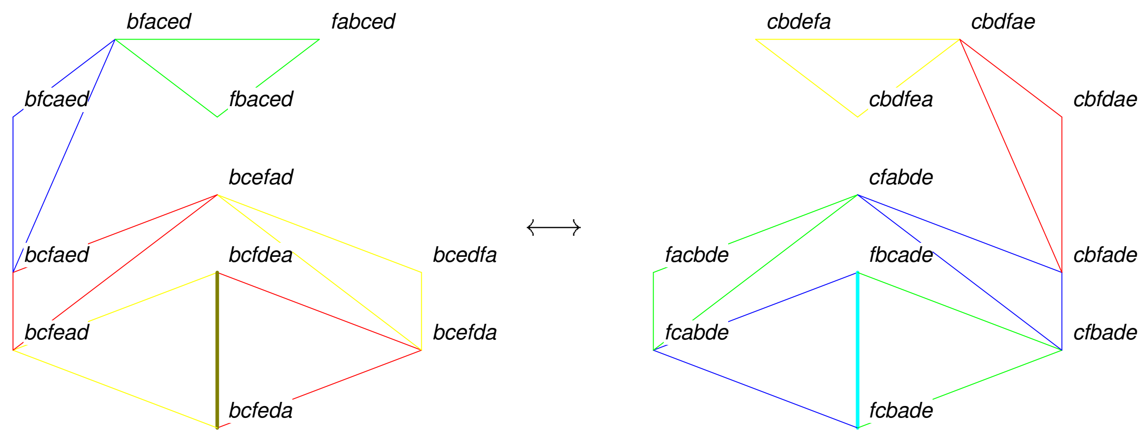


Figure 2: Schützenberger-equivalent classes: *facbde* and *bcedfa*

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:

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

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

$$u \equiv v \quad (CH(A,<)) \quad \Rightarrow \quad Std(u) \equiv Std(v) \quad (CH(A \times \mathbb{N},<)), \tag{8}$$

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

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

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

The standardization process shall enable us to extend results proved for standard words to general words.

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:

$$u \equiv v \quad \Rightarrow \quad \chi_I(u) \equiv \chi_I(v) \; . \tag{9}$$

We just have to check property (9) 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

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

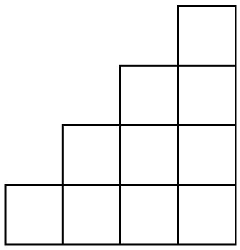
which is obviously satisfied. This concludes the proof of (9).

This property, a kind of converse of Standardization, is also used to transfer results from standard to general words.

2 A representation of the Chinese monoid

2.1 Chinese staircases

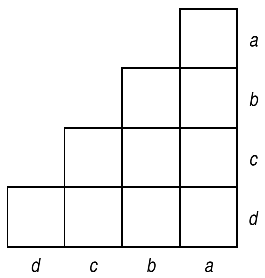
Let A be a totally ordered alphabet of n letters, let $p \leq n$. A staircase diagram over A is a Ferrers diagram of shape $(1, 2, \ldots, n)$ that we draw in the following way:



the example being given here for $n = 4$. A *Chinese staircase* (or plain Chinese staircase for emphasis) is a staircase diagram filled with nonnegative integers,[2] e.g.,

			2
		1	0
	3	2	2
2	1	0	1

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



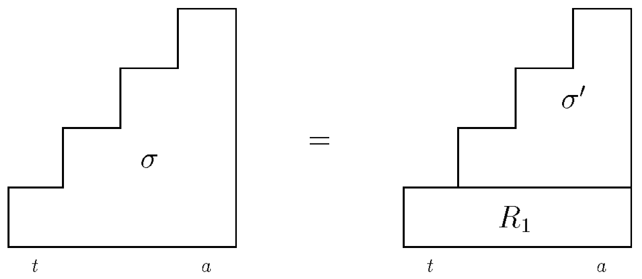
this example being for $A = \{a, b, c, d, \}$, $n = 4$.

We denote indifferently by $\sigma_{\alpha\beta}$ the cell in row α, column β or its contents. Formally, a Chinese 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.

Let $a, b, \ldots, s$ be the first p letters of the alphabet. A *partial Chinese staircase*, or p-Chinese staircase for short, is a Chinese staircase over a n letters alphabet such that all cells below row p are empty: $\forall \alpha > s, \quad \forall \beta \in A, \quad \sigma_{\alpha\beta} = 0$. We say that s is the greatest letter of the p-Chinese staircase σ, which just means that σ is empty below the row of s.

Note 2.1 We need a form of structural induction on staircases. Let σ be a p-staircase of greatest letter s. Let row R be an application from $a, b, \ldots, s, t$ into $\mathbb{N}$. We denote by $\sigma' = (\sigma,\ R)$ the $p+1$-staircase built from σ by setting $\sigma'_{t\alpha} = R(\alpha)$.

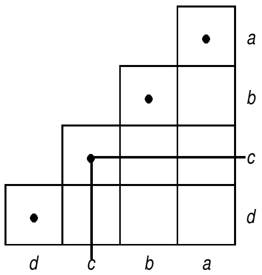
Conversely, if σ is a p-staircase with $p > 0$, let R_1 be the row p of σ, σ' the $p-1$-staircase obtained by setting all the cells of row p of σ to 0. Then $\sigma = (\sigma',\ R_1)$.



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

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

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's 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

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

where $a, b, \dots, z$ 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

σ_z	σ_{zy}	$\cdots$	σ_{za}
z	y		a

,

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

$$w = l_a \, l_b \dots \, l_z \tag{10}$$

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

Definition 2.5 *A word w is the row normal form of a staircase σ if w is the Chinese staircase word obtained by concatenating all Chinese rows associated with the rows of σ from top to bottom. We write $w = r(\sigma)$.*

We will define later (in Definition (2.18)) the column normal formal of a Chinese staircase as a dual notion. For practical use, we can read the row normal form on the staircase directly, from top to bottom, row by row, and from right to left in each row.

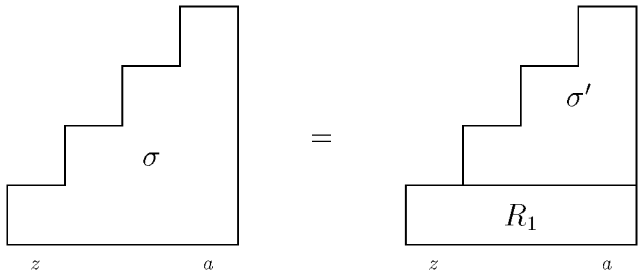
2.2 The Insertion Algorithm

We shall now describe an algorithm that inserts a word of A^* into a Chinese staircase. The basic idea is to try to adapt Schensted's algorithm (Schensted [14], Knuth [4], Knuth [5]). The basic algorithm builds upon a Chinese staircase σ and a letter α a new Chinese staircase denoted by $\sigma.\alpha$. From there, we denote by $\sigma.a_1 a_2 \dots a_k$ the staircase obtained by inserting successively a_1, a_2, $\dots$, a_k into σ, i.e., $\sigma.(a_1 a_2 \dots a_k) = (\cdots((\sigma.a_1).a_2)\cdots).a_k$.

Algorithm 2.6 (The insertion algorithm)
This algorithm describes a recursive procedure that inserts a letter α into a p-Chinese staircase σ.

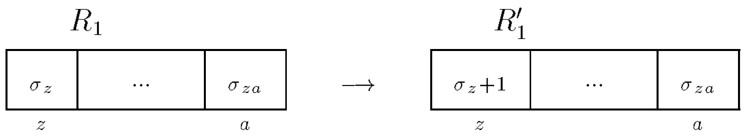
The insertion of a letter into a plain Chinese staircase simply corresponds to the special case where $p = n$.

Write $\sigma = (\sigma',\ R_1)$, where R_1 is the bottom row of σ; let z be the greatest letter of σ.



1. If $\alpha > z$, then the algorithm ends with an error.

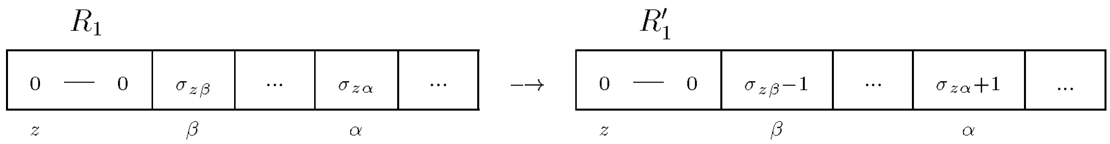
2. If $\alpha = z$, then $\sigma.\alpha = (\sigma',\ R_1')$ where R_1' is obtained from R_1 by adding 1 to cell σ_z:



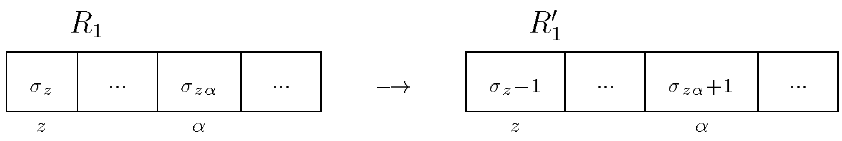
3. If $\alpha < z$, let β be the greatest letter whose cell on R_1 does not contain 0; if such a β does not exist, set $\beta = \alpha$. Three distinct cases appear:

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

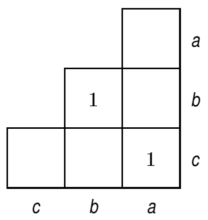
   3b. If $\alpha < \beta < z$, then $\sigma.\alpha = (\sigma'.\beta,\ R_1')$, where R_1' is obtained from R_1 by adding 1 to cell $\sigma_{z\alpha}$ and substracting 1 from cell $\sigma_{z\beta}$:



   3c. 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:



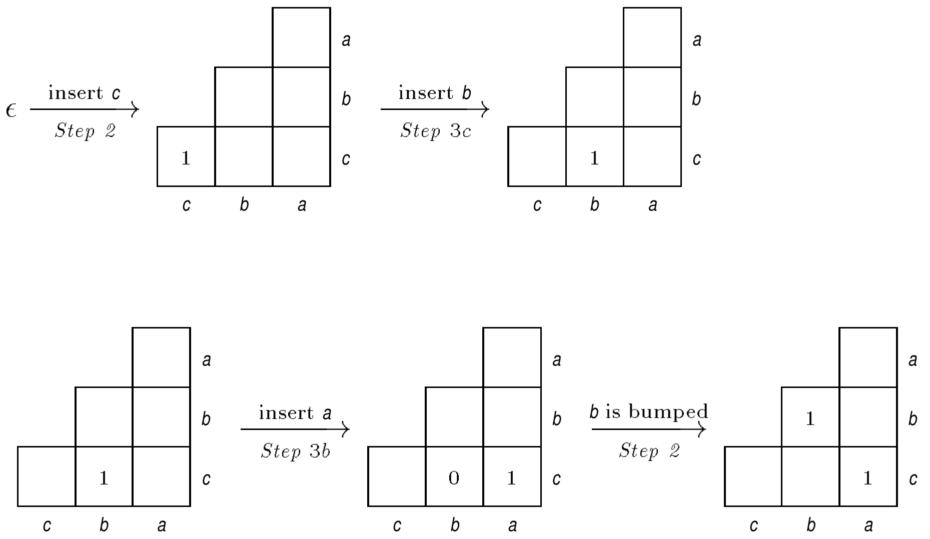
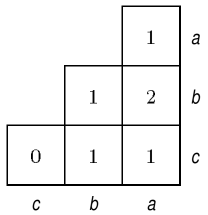


Figure 3: The insertion of *cba*

Figure 3 shows the insertion algorithm in action.
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 explicitly all the steps in the insertion of a letter $\alpha = \alpha_1$. The insertion algorithm deals with a decreasing sequence of partial staircases:*

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

The sequence $\alpha_1 \xrightarrow{s_1} (R'_1, \alpha_2) \xrightarrow{s_2} (R'_2, \alpha_3) \cdots (R'_{k-1}, \alpha_k) \xrightarrow{s_k} R'_k$ describes the execution of the algorithm accurately.[3] *We call such a sequence an* insertion sequence*. When the emphasis is upon the letters and the corresponding transitions, but we don't care about the actual rows, nor the whole sequence, we will also use some relaxed notation, e.g., $\alpha_2 \xrightarrow{s_2} \alpha_3$.*

[3] In the semantics field, this is a common notion, usually called a *trace* of the algorithm. The intuition is that the analysis of an algorithm can be simplified by first observing selected behavior.

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 $\gamma \geq \alpha$ and $\delta > \beta$. An* exposed letter *is a letter that indexes a column corresponding to an exposed entry.*

Proposition 2.10 *For a Chinese staircase σ, the insertion algorithm yields an increasing sequence of letters $\alpha_1, \alpha_2, \ldots \alpha_k$ that takes as values all the exposed letters of σ 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 can never occur for an insertion in a plain Chinese 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:*

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

Definition 2.12 *Let ϵ be the empty staircase over A. 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 class of w. Theorem 2.14, to follow, fully justifies the use of a similar notation.*

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. By inspection of Step 1 to Step 3c, checking that $|\sigma.\alpha|_\alpha = |\sigma|_\alpha + 1$, and that $|\sigma.\alpha|_\beta = |\sigma|_\beta$, for $\beta \neq \alpha$ is immediate. 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, the insertion algorithm yields the same staircase σ for v and w.*

Proof —
Property 1: First we check that $r(\sigma)$ yields the staircase σ, so that Property 1 simplifies to Property 1′: for any word w for which the insertion algorithm yields the staircase σ, w is congruent to $r(\sigma)$.

Lemma 2.15 *Let σ be a staircase and $t = r(\sigma)$. 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, σ the staircase obtained by inserting w, and $t = r(\sigma)$. The proof is obtained by induction on the number of letters of w. If $|w| \leq 2$, there is nothing to prove. If $|w| \geq 3$, assume Property 1′ holds for any word w', $|w'| < |w|$.

Decompose $w = w_1\alpha$ with $\alpha \in A$. Property 1′ holds for w_1: if σ_1 is the staircase obtained by the insertion of w_1, the row normal form $t_1 = r(\sigma_1)$ 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. In parallel with the algorithm, we perform a structural induction on the staircase σ_1.

Let σ_0 be a partial staircase of σ_1, let t_0 be the row normal form of σ_0. Decompose $\sigma_0 = (\sigma_0', R_1)$ and $t_0 = t_0'r$ with r the Chinese row of type z associated with R_1. The induction hypothesis states that, for any β smaller than or equal to the greatest letter of σ_0', t_0' concatenated with β is congruent to the row normal form of $\sigma_0'.\beta$. Let z be the greatest letter of σ_0 and keep in mind that $r \in (za)^*(zb)^* \ldots (zy)^* z^*$ and insert α in r:

– Step 1: Not applicable.

– Step 2: $\sigma_0.\alpha = (\sigma_0', R_1')$, with $r' = rz$ the Chinese row associated with R_1'. No rewriting is necessary.

– Step 3a: By applying the insertion algorithm, we obtain $\sigma_0.\alpha = (\sigma_0'.\alpha, R_1)$. Since $\beta \leq \alpha < z$, $r \in (za)^*(zb)^* \ldots (zy)^*$. 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 $(z\gamma)$ occurring in r, $\gamma \leq \beta \leq \alpha$, so that $(z\gamma)\alpha \equiv \alpha(z\gamma)$: α crosses through each pair of letters $(z\gamma)$.

– Step 3b: By applying the insertion algorithm, we obtain similarly $\sigma_0.\alpha = (\sigma_0'.\beta, R_1')$. Since $\beta < z$, $r \in (za)^*(zb)^* \ldots (z\beta)^*$. The induction hypothesis applies since $\beta < z$, so we just have to verify that $r\alpha \equiv \beta r'$, where r' is the Chinese word associated to R_1'.

  – 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 $(z\gamma)$ occurring in r, $(z\gamma)\beta \equiv \beta(z\gamma)$: β crosses through each pair of letters $(z\gamma)$, as in Step 3a.

  – For every $(z\gamma)$ occurring 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 $2'$:

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

for every elementary congruence $a \leq b \leq c$. 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,

$$\begin{array}{|c|c|}\cline{2-2} \multicolumn{1}{c|}{} & \sigma_a \\ \hline \sigma_b & \sigma_{ba} \\ \hline \end{array}\,.bba \quad = \quad \begin{array}{|c|c|}\cline{2-2} \multicolumn{1}{c|}{} & \sigma_a \\ \hline \sigma_b{+}1 & \sigma_{ba}{+}1 \\ \hline \end{array} \quad = \quad \begin{array}{|c|c|}\cline{2-2} \multicolumn{1}{c|}{} & \sigma_a \\ \hline \sigma_b & \sigma_{ba} \\ \hline \end{array}\,.bab.$$

Let σ be a p-staircase of greatest letter z. Let a, b, c be three letters such that $a \leq b \leq c \leq z$. 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 $a < a' \leq b$. 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':

$$\begin{array}{|c|c|c|c|c|l}\cline{1-5} 0 \;—\; 0 & \sigma_{za'} & \cdots & \sigma_{za} & \cdots & R \\ \cline{1-5} \multicolumn{6}{c}{} \\ \cline{1-5} 0 \;—\; 0 & \sigma_{za'}-1 & \cdots & \sigma_{za}+1 & \cdots & R' \\ \cline{1-5} \multicolumn{1}{c}{z \quad b} & \multicolumn{1}{c}{a'} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{a} & \multicolumn{1}{c}{} & \end{array}$$

   1.3. The exposed letter in R is b' with $b < b' \leq c$. 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 $b' \geq a' \geq b$. 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 $d \geq b$, 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 $c \leq c'$.

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

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 $d \leq b$. 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. $\square$

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 normal form of a staircase as we defined the row normal form in Section 2.1. A word w is a *Chinese column* of type a if and only if it has the following structure

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

where $a, b, \dots, z$ 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

$$w = l'_a \, l'_b \dots \, l'_z \tag{11}$$

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

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. We write $w = c(\sigma)$.*

Proposition 2.19 *For any staircase σ, $r(\sigma)$ and $c(\sigma)$ are equivalent with respect to the Chinese congruence.*

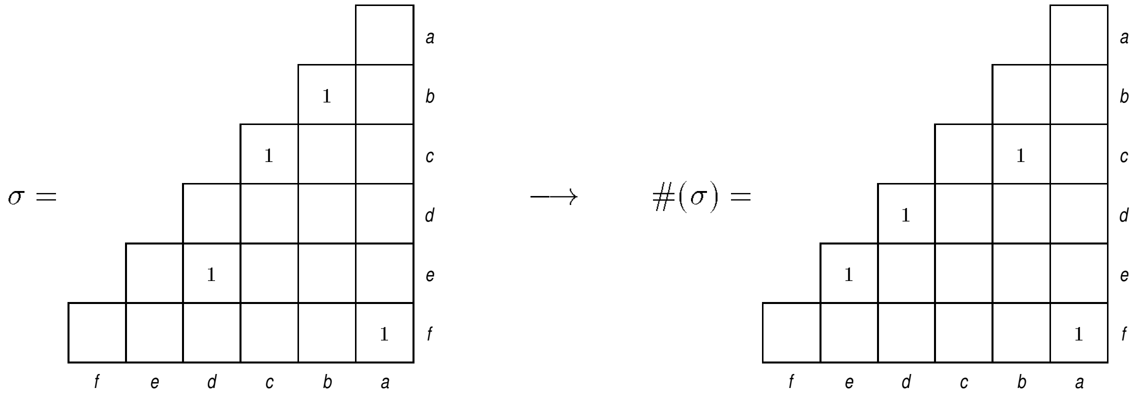
Proof — It is clear that σ is the staircase obtained by the insertion algorithm applied to $c(\sigma)$. The conclusion then follows from Property 1.

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 $A = \{a, b, \dots, z\}$ as the line of cells σ_{az}, $\sigma_{by}, \dots$

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

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

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 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 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 $\epsilon.a_1.\ldots.a_k$.

Definition 3.2 *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 3.3 (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 $\sigma \xrightarrow[\delta]{} \tau$ 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 $\tilde{\sigma} \xrightarrow[\tilde{\delta}]{} \tilde{\tau}$, 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 $(\sigma \xrightarrow[\delta]{} .)$ as follows:

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

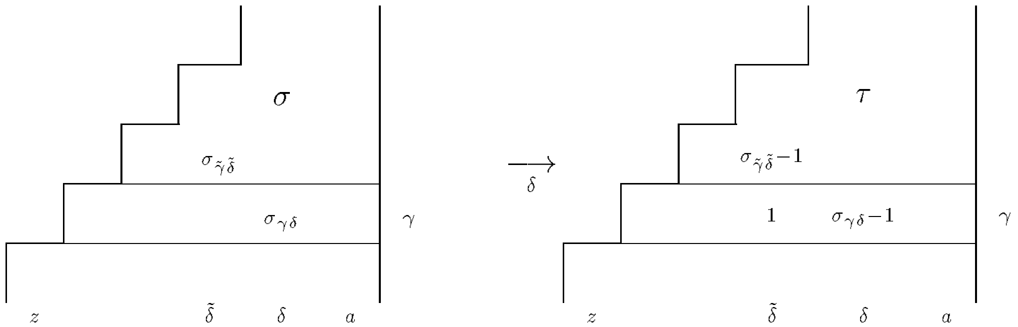


Figure 4: How Rule $2_{\tilde{\delta}}$ works.

Example 3.4 Consider the Chinese row staircase word *cbdfega*. We display in Figure 5 the staircases obtained from *cbdfega* by applying Algorithm 3.3.

We now prove that this algorithm is correct.

Theorem 3.5 *Let σ be a standard staircase over n letters. Define:*

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

Then

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

– *$\sigma.\alpha^{-1}$ is equal to $(\sigma \xrightarrow[\alpha]{} .)$ for any α.*

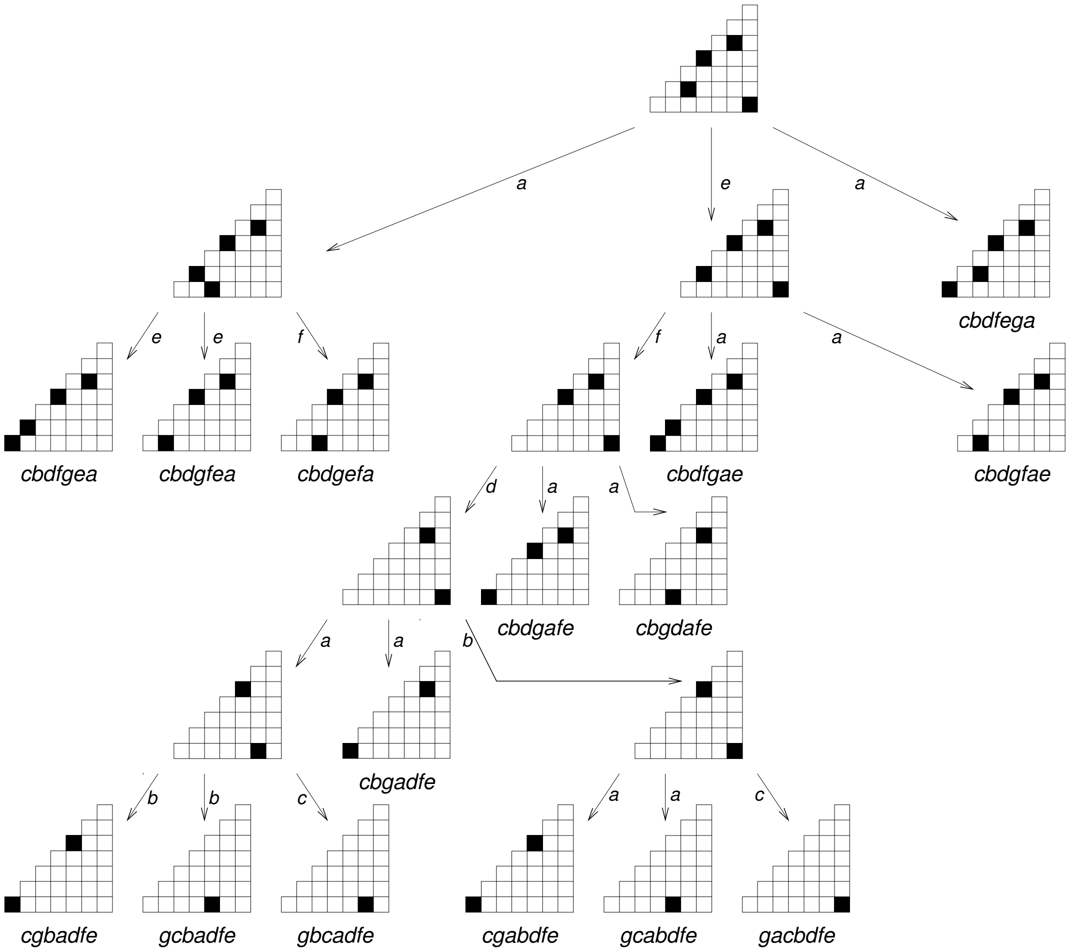


Figure 5: The cardinal of the class of *cbdfega* is 15.

First, we prove that $\Delta(\sigma)$ 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. □

Then we prove that $\sigma.\alpha^{-1}$ is equal to $(\sigma \underset{\alpha}{\longrightarrow} .)$ for any α.

Proof — The proof proceeds by induction on the size of the staircase.
First, we show that $(\sigma \underset{\alpha}{\longrightarrow} .) \subset \sigma.\alpha^{-1}$. In fact, we have just proved the case of Rule 1 and of Rule 1′. Assume the notations of Rule $2_{\tilde{\delta}}$. Thanks to the induction hypothesis, $\tilde{\tau}.\tilde{\delta} = \tilde{\sigma}$. A close examination of Step $3b$ 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

[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.

not constant, the insertion sequence has to contain a step $\alpha = \alpha_i \xrightarrow{3b} \alpha_{i+1} = \beta$. By the induction hypothesis, $\sigma^{(i)}.\beta \underset{\beta}{\longrightarrow} \sigma^{(i)}$, and therefore, $\sigma.\alpha \underset{\alpha}{\longrightarrow} \sigma$ by Rule 2_β. □

3.2 Applications

Theorem 3.6 *In the standard case, every class has an odd order.*

Proof — Let

$$\sigma.A^{-1} = \bigcup_{\delta \in \Delta(\sigma)} \sigma.\delta^{-1}.$$

The union $C(\sigma.A^{-1})$ 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, it suffices 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 move down the staircase in considering deletable entries, 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 3.7 *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 3.5, 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 3.8 *$\sigma_\perp$ is the smallest staircase of $\sigma.A^{-1}$.*

Proof — Let σ' be a staircase obtained by applying one rewriting rule 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 $\sigma_\perp < \sigma'$. □

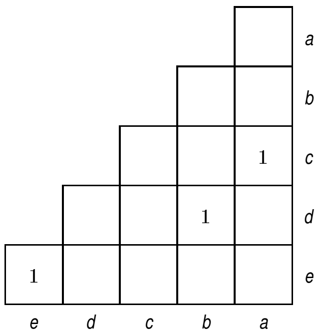
We can then deduce that the smallest element is obtained by removing letters in each row from bottom to top and in a row from left to right (in the non standard case.) We thus obtain the row normal form of σ. □

3.3 Link Representation

Definition 3.9 *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 obtained 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* $\lambda(\sigma) =$ 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:

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

Definition 3.10 *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 3.11 Take the class of abdc, of link representation abcd. That is, a and b are neutral, c is small, d is great.

4 The Great Class

We devote this section to several counting results concerning standard Chinese classes. First we characterize the largest class of the Chinese monoid. This leads to the construction of a correspondence between the words of this class and Dyck words.

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 Recall Definition 3.10 and let $h = [\frac{n}{2}]$. In $Gr(n)$, the great letters are the h greatest letters of the alphabet and the small letters are the h smallest. If n is odd the $(h+1)^{\text{th}}$ 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.

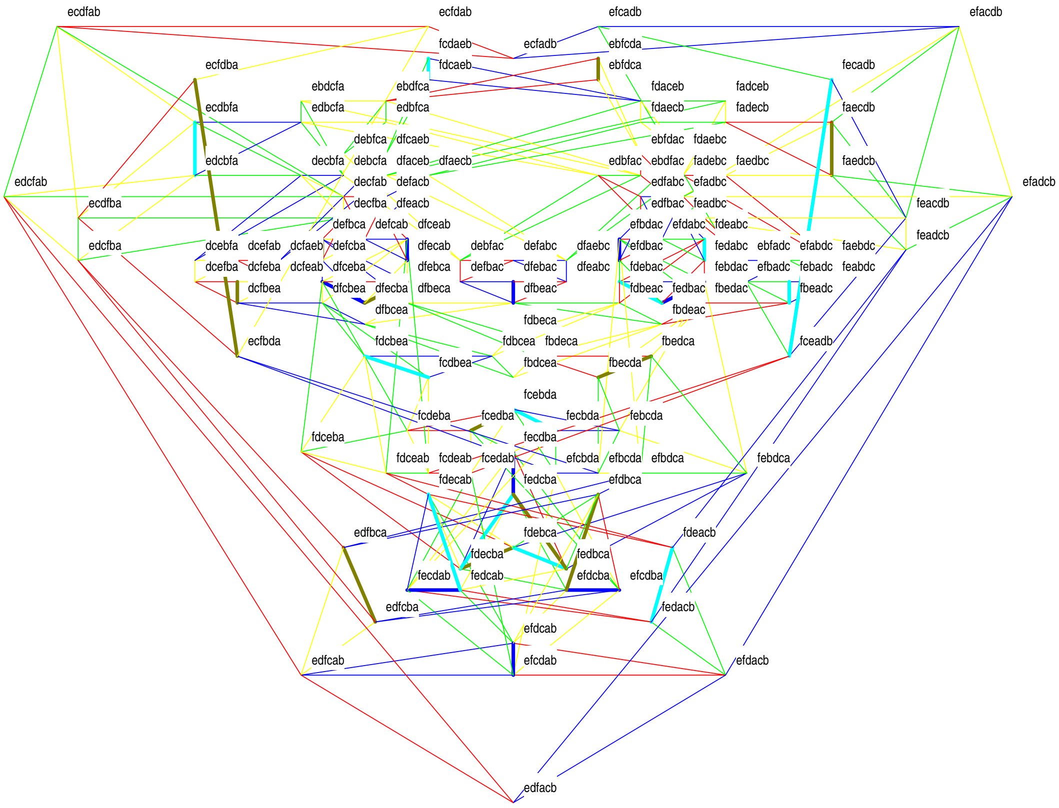


Figure 6: The great class *fedcba*: $\lambda($*dcebfa*$) =$ *abcdef*.

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.
- If $\alpha \xrightarrow{3a} \beta$, then α replaces and dislodges 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 an intuitive 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 dislodge 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 congruence at position k between u, v, w is an elementary congruence of the Chinese relation: there exists three letters a, b, c, a word x of length k, another word 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 congruence between words u, v, w of C_1, there exists an elementary congruence between $i(u)$, $i(v)$, $i(w)$ that occurs 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 larger class. Therefore, each class belongs to a sequence of increasing classes that ends necessarily at the great class.

Algorithm 4.5 (Elementary Class Embedding)
Let σ be a full standard staircase such that $C(\sigma)$ is not the great class. 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 and denote by t_β^γ the elementary tranposition which exchanges β and γ and leave other letters invariant. The algorithm yields 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 an 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 possible cases to consider: $\beta < \gamma \leq c$, so that $c\gamma\beta \equiv c\beta\gamma \equiv \gamma c\beta$ and $a \leq \beta < \gamma$, so that $\gamma\beta a \equiv \gamma a\beta \equiv \beta\gamma a$.

We contend that no word of $C(\sigma)$ contains such a sequence. More precisely, γ occurs after β in every word of $C(\sigma)$. 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 that 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 occurs after β in every word of $C(\sigma)$. Every elementary congruence of $C(\sigma)$ 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 an embedding of this class into $Gr(n)$. Moreover, $Gr(n)$ is strictly larger than any other Chinese class.*

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 an embedding of $C(\sigma_i)$ into $C(\sigma_j)$.

Each elementary embedding injects a Chinese class $C(\sigma_i)$ into a Chinese class $C(\sigma_{i+1})$ of greater or equal order. We have to prove that this process must end. 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 holds the greatest possible number of inversions, that is $u_j = \omega$, or such that the algorithm fails, that is, $u_j \in Gr(n)$.

More precisely, we can formulate a finer analysis of the Elementary Class Embedding Algorithm. Recalling the proof of proposition 4.6, we did prove that words of the source class always yields β before γ. Obviously, the image of the source class through t_β^γ always yield γ before β. Looking at the possible elementary congruences that involve β and γ, there are two cases to consider. Either no word of the source class holds a sequence of the form $\beta c\gamma$ or $\beta a\gamma$, in which case t_β^γ only puts β and γ in the right order, or the source class holds such a sequence, in which case t_β^γ 'unlocks' a new elementary congruence, and embeds the class into a strictly larger class.

We now return to the sequence of successively larger Chinese classes $C(\sigma_0), \ldots, C(\sigma_{j-1})$, $C(\sigma_j) = Gr(n)$. Consider the last transposition t_b^c which sends $C(\sigma_{j-1})$ into $Gr(n)$. Since ω belongs to $Gr(n)$, this word contains a sequence of type $\gamma\beta a$ or $c\gamma\beta$, and by extension $Gr(n)$ contains the whole set of elementary congruences, $c\gamma\beta \equiv c\beta\gamma \equiv \gamma c\beta$ (resp. $\gamma\beta a \equiv \gamma a\beta \equiv \beta\gamma a$). Going back to $C(\sigma_{j-1})$, this class cannot contain the transposed pattern $c\gamma\beta$ (resp. $\gamma\beta a$), hence $c\beta\gamma$ (resp. $\beta\gamma a$) cannot have an antecedent: $Gr(n)$ is strictly larger. □

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.

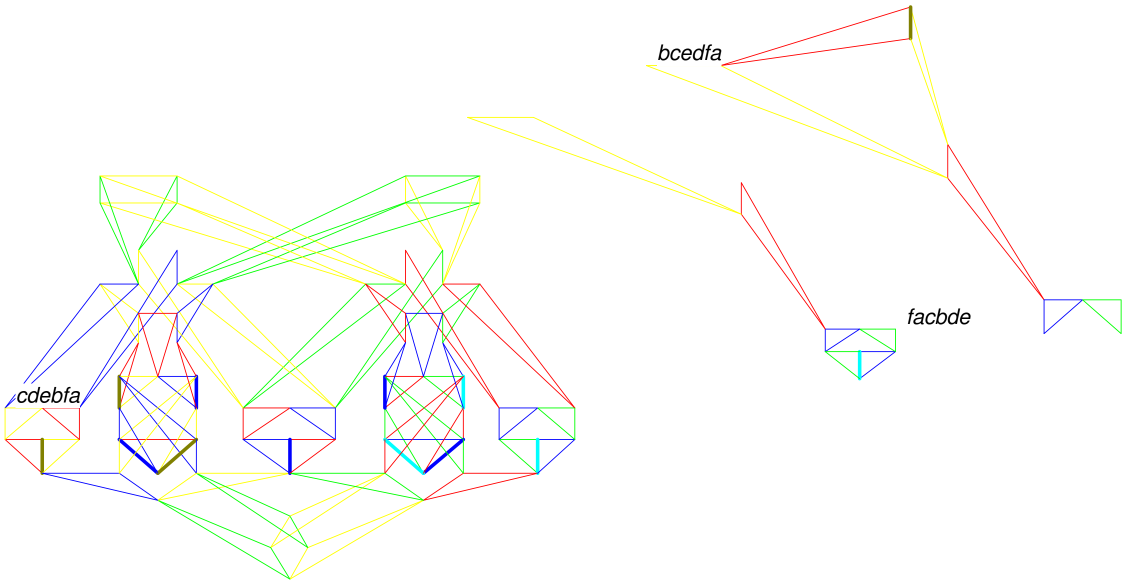


Figure 7: The transposition of *c* and *d* embeds the class of *cdebfa* (order 75) into the class $Gr(6)$ (order 135). Note that the embeddings of *facbde* and *bcedfa* do not preserve Schützenberger equivalence.

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 $n = 2h$ 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 exactly one return to zero.

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

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

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

- *if w belongs to $Gr(2h)$, if w' is obtained by deleting c and c' from w, then w' does belong to $Gr(2h-2)$.*
- *if w' belongs to $Gr(2h-2)$, 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 $Gr(2h)$.*
- *if w' belongs to $Gr(2h-2)$, 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 $Gr(2h)$.*

For example, **faeb** $\in Gr(4)$ and $\pi($**faeb**$) = x\bar{x}x\bar{x} \in D_2$. 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]

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.

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

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, denoted by 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, denoted by B_i (resp. B'_i), is the remaining set of 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.

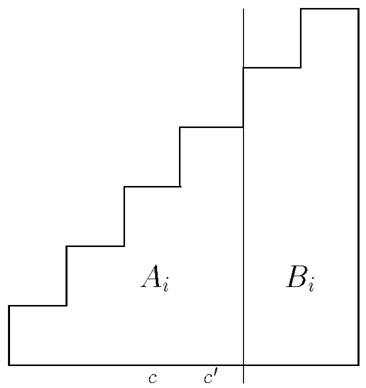


Figure 8: The definition of A_i and B_i.

One direct 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 $Gr(2h-2)$ class. We have to check where we can insert c and c' in a word of $Gr(2h-2)$ to obtain a word of $Gr(2h)$. 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 precisely the number of letters remaining in the left part of the staircase A'_i. Let $i = \inf\{i > j \mid \quad h(a_1 \dots a_i) = 0\}$.

We now have a rather precise picture of the sequence of staircases σ'_i: the height of the Dyck word corresponding to w' accounts for the number of small letters still to be inserted in the staircase before the left part A'_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 $Gr(2h)$ 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 $Gr(2h)$ 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 $Gr(2h)$ 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 $Gr(2h)$ 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 $Gr(2h)$ and the Dyck words with history of length $2p$.

Actually, we can count the words of $Gr(2h)$ 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 $\cancel{x}x\cancel{\bar{x}}\bar{x}$, $\cancel{x}x\bar{x}\cancel{\bar{x}}$, $x\cancel{x}\cancel{\bar{x}}\bar{x}$, and $x\cancel{x}\bar{x}\cancel{\bar{x}}$, $Weight(xx\bar{x}\bar{x}) = 4$. On the other hand, $x\bar{x}x\bar{x}$ reduces as $\cancel{x}\cancel{\bar{x}}x\bar{x}$, $x\cancel{\bar{x}}\cancel{x}\bar{x}$, and $x\bar{x}\cancel{x}\cancel{\bar{x}}$, 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 $2h+1$ letters. Denote by c the median letter of A; model $Gr(2h)$ over $A\setminus\{c\}$. Then the great class $Gr(2h+1)$ is characterized as follows:*

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

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

5 Conjugacy classes

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

Definition 5.1 *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 5.2 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 5.5, to follow.

Theorem 5.3 *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 5.4 *We define yet another equivalence relation on words:*
$a_1a_2\ldots a_k = w \approx w' = a'_1a'_2\ldots a'_k$ if and only if:

- *$w \equiv w'$ or*
- *$a_{k-1}a_ka_1 \equiv a'_{k-1}a'_ka'_1$ or*
- *$a_ka_1a_2 \equiv a'_ka'_1a'_2$,*

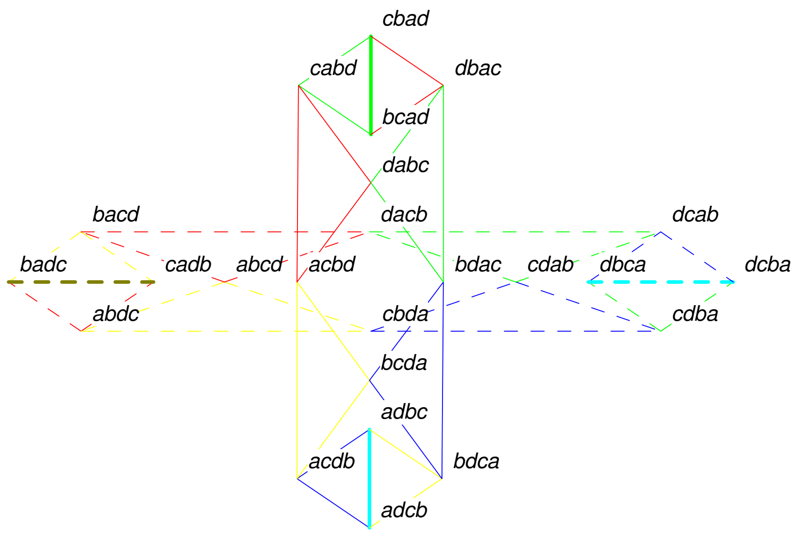


Figure 9: the wrapping Chinese congruence over 4 letters

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 9 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 5.5 *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:

Algorithm 5.6 (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 $\approxeq$: $cba \approxeq cab$ and $bca \approxeq cab$ (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 $y(za) \approxeq (za)y$.[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 5.6 to w clearly yields

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

– either $w \approx zayb \ldots \alpha\beta$,

– or $w \approx zayb \ldots \beta\alpha$,

with α and β the median letters of A. If n is odd, iterated application of Algorithm 5.6 yields $zayb \ldots \alpha$ 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 5.6 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. □

Note 5.7 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 5.8 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 developed a suite of programs 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, [16]). 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 correspondence*, [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] Krob D., Thibon J.-Y., *Noncommutative symmetric functions IV: Quantum linear groups and Hecke algebras at q=0*, J. Alg. Comb. (to appear)

[7] Krob D., Thibon J.-Y., *Noncommutative symmetric functions V: A crystalizable version of* $U_q(\mathcal{G}l_N)$, [in FPSAC'97, C. Krattenthaler, Ed.], 326–338, 1997

[8] 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.

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

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

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

[12] Novelli J.-C., *On the hypoplactic monoid*, [in FPSAC'97, C. Krattenthaler, Ed.], 460–472, 1997

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

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

[15] 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.

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