Finite Gröbner–Shirshov bases for Plactic algebras and biautomatic structures for Plactic monoids

Alan J.Cain, Robert D. Gray, António Malheiro

[AJC] Centro de Matemática, Faculdade de Ciências, Universidade do Porto
Rua do Campo Alegre 687, 4169–007 Porto, Portugal

Email: ajcain@fc.up.pt
Web page: www.fc.up.pt/pessoas/ajcain/

[RDG] Centro de Álgebra da Universidade de Lisboa
Av. Prof. Gama Pinto 2, 1649–003 Lisboa, Portugal

Email: Robert.D.Gray@uea.ac.uk

[AM] Centro de Álgebra da Universidade de Lisboa
Av. Prof. Gama Pinto 2, 1649–003 Lisboa, Portugal
   and
Departamento de Matemática,
Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa,
2829–516 Caparica, Portugal

Email: ajm@fct.unl.pt

ABSTRACT

This paper shows that every Plactic algebra of finite rank admits a finite Gröbner–Shirshov basis. The result is proved by using the combinatorial properties of Young tableaux to construct a finite complete rewriting system for the corresponding Plactic monoid, which also yields the corollaries that Plactic monoids of finite rank have finite derivation type and satisfy the homological finiteness properties left and right FP_∞. Also, answering a question of Zelmanov, we apply this rewriting system and other techniques to show that Plactic monoids of finite rank are biautomatic.

Keywords: Plactic algebra; Plactic monoid; Gröbner–Shirshov basis; complete rewriting system; Young tableau; automatic monoids.

2010 Mathematics Subject Classification: 16s36; 68Q42, 20M25, 20M35.

1 INTRODUCTION

The Plactic monoid has its origins in work of Schensted [Sch61] and Knuth [Knu70] concerned with certain combinatorial problems and operations on Young tableaux. It was later studied in depth by Lascoux and Shützenberger [LS81] and has since become an important tool in several aspects of representation theory and algebraic combinatorics; see [Ful97, Lot02]. The first significant application of the Plactic monoid was to the Littlewood–Richardson rule for Schur functions. This is explained in detail in the appendix to the second edition of J. A. Green's influential

monograph on the representation theory of the general linear group [Gre07]. The Littlewood–Richardson rule [LR34] is one of the most important results in the theory of symmetric functions. It provides a combinatorial rule for expressing a product of two Schur functions as a linear combination of Schur functions. Since Schur functions in n variables are the irreducible polynomial characters of $GL_n(\mathbb{C})$, the Littlewood-Richardson rule gives a tensor product rule for $GL_n(\mathbb{C})$. One of the most enlightening proofs of the Littlewood–Richardson rule (see [Lot02, Section 5.4]) is given by lifting the calculus of the Schur function to the integral monoid ring of the Plactic monoid (called the tableau ring; see [Ful97, Chapter 2]).

Subsequently the Plactic monoid has been found to have applications in a range of areas including a combinatorial description of Kostka–Foulkes polynomials [LS81, LS78], and to Kashiwara's theory of crystal bases [DJM90, Kas91] leading to the definition of Plactic algebras associated to all classical simple Lie algebras [Lit96, LLT95, KT97]. Further results on Robinson–Schensted correspondence and the Plactic relations may be found in [DJM90, LT96]. Several variations and generalizations of the Plactic monoid have been proposed and investigated including hypoplactic monoids [KT97], and shifted Plactic monoids [Ser10]. In [DK94] it is show that the Hilbert series of the Plactic monoid is given by the Schur–Littlewood formula, and that there are exactly three families of ternary monoids with this Hilbert series. Schützenberger [Sch97] argues that the Plactic monoid ought to be considered as "one of the most fundamental monoids in algebra". He cites three reasons for his own personal "weakness" for the Plactic monoid, the first of them being the application to symmetric functions mentioned above.

Various aspects of the corresponding semigroup algebras, the Plactic algebras, have been investigated; see, for example, [CO04, LS90]. These algebras are important special cases in the more general study of algebras defined by homogeneous semigroup presentations [CJO10]. Frequently, fundamental problems about such semigroup algebras require detailed analysis of the corresponding semigroups. An important example of this is given by the theory of Gröbner–Shirshov bases. Kubat & Okniński showed that the Plactic algebra of rank 3 has a finite Gröbner–Shirshov basis [KO14, Theorem 1] and that Plactic algebras of rank 4 or more do not admit a finite Gröbner–Shirshov basis with respect to the degree-lexicographic ordering over the usual generating set for the Plactic monoid [KO14, Theorem 3]. In contrast, the related Chinese monoid admits a finite complete rewriting system with respect to the usual generating set [GK10], and so its semigroup algebra, the Chinese algebra, is known to admit a finite Gröbner–Shirshov basis [CQ08].

The first aim of this paper is to use the combinatorial properties of Young tableaux to construct finite complete rewriting systems for Plactic monoids of arbitrary finite rank, and thus prove that the corresponding Plactic algebras admit finite Gröbner–Shirshov bases (see [Hey00] for an explanation of the connection between Gröbner–Shirshov bases and complete rewriting systems). The rewriting system is *not* over the usual generating set for the Plactic monoid; rather, the generating set comprises the (finite) set of columns of Young tableaux. As a corollary we deduce that Plactic monoids of finite rank satisfy the homological finiteness property FP_∞, a result which gives information about the existence of free resolutions of $\mathbb{Z}M_n$-modules, where $\mathbb{Z}M_n$ is the tableau ring featuring in the theory of symmetric functions outlined above.

During the writing of this paper, the authors came across the work of Chen & Li [CL11], who exhibit infinite complete rewriting systems for Plactic monoids over the (infinite) set of rows of Young tableaux. Thus Chen & Li's work yields infinite Gröbner–Shirshov bases for Plactic algebras. Part of their reasoning is an analogue for rows of Lemma 3.1 below, but they use a direct, more technical, proof and later recover as a corollary of their main result the fact that tableaux form a cross-section of the Plactic monoid.

As a consequence of the Schensted insertion algorithm and the representation of elements by tableaux, it follows that the Plactic monoid has word problem that is solvable in quadratic time. This leads us naturally to the second major theme of the present article: the subject of automatic structures. The concept of an automatic group

was introduced in order to describe a large class of groups with easily solvable word problem. The best general reference for the theory of automatic groups is the book [ECH⁺92]. The notion has been extended to automatic monoids and semigroups [CRRT01]. In both cases the defining property is the existence of a rational set of normal forms (with respect to some finite generating set A) such that we have, for each generator in A, a finite automaton that recognizes pairs of normal forms that differ by multiplication by that generator. It is a consequence of the definition that automatic monoids (and in particular automatic groups) have word problem that is solvable in quadratic time [CRRT01, Corollary 3.7].

Automatic groups have attracted a lot of attention over the last 20 years, in part because of the large number of natural and important classes of groups that have this property. The class of automatic groups includes: finite groups, free groups, free abelian groups, various small cancellation groups [GS90], Artin groups of finite and large type [HR12], Braid groups, and hyperbolic groups in the sense of Gromov [Gro87]. In parallel, the theory of automatic monoids has been extended and developed over recent years. Classes of monoids that have been shown to be automatic include divisibility monoids [Pic06] and singular Artin monoids of finite type [CHKT11]. Several complexity and decidability results for automatic monoids are obtained in [Loh05]. Other aspects of the theory of automatic monoids that have been investigated include connections with the theory of Dehn functions [Ott00] and complete rewriting systems [OSKM98].

Given the algorithmic properties of the Plactic monoid mentioned above, the natural question of whether the Plactic monoid itself admits an automatic structure was asked by Efim Zelmanov [during his plenary lecture at the international conference *Groups and Semigroups: Interactions and Computations* (Lisbon, 25–29 July 2011)]. The second main result of this article is an affirmative answer to this question. Beginning with the finite complete rewriting system obtained in Section 3, we shall show how for Plactic monoids finite transducers may be constructed to perform left (respectively right) multiplication by a generator. We then apply this result to show that Plactic monoids of arbitrary finite rank are biautomatic (the strongest form of automaticity for monoids).

2 PRELIMINARIES

This paper assumes familiarity with rewriting systems, Gröbner–Shirshov bases, automata and regular languages, and transducers and rational relations.

For background information, see, for example, [BO93] on complete rewriting systems; [Ufn98] on Gröbner–Shirshov bases; [Hey00] on the connection between them. See also [HU79] on automata and regular languages and [Ber79]) on transducers and rational relations.

We denote the empty word (over any alphabet) by ε. For an alphabet A, we denote by A^* the set of all words over A. When A is a generating set for a monoid M, every element of A^* can be interpreted either as a word or as an element of M. For words $u, v \in A^*$, we write $u = v$ to indicate that u and v are equal as words and $u =_M v$ to denote that u and v represent the same element of the monoid M. The length of $u \in A^*$ is denoted $|u|$. For a relation $\mathcal{R}$ on A^*, the presentation $\langle A \mid \mathcal{R} \rangle$ defines [any monoid isomorphic to] $A^*/\mathcal{R}^\#$, where $\mathcal{R}^\#$ denotes the congruence generated by $\mathcal{R}$.

2.1 *Plactic monoid*

This section recalls only the relevant definition and properties of the Plactic monoid; for a full introduction, see [Lot02, Chapter 5].

Let $n \in \mathbb{N}$. Let A be the finite ordered alphabet $\{1 < 2 < \ldots < n\}$. Let $\mathcal{R}$ be the set of defining relations

$$\{(xzy, zxy) : x \leqslant y < z\} \cup \{(yxz, yzx) : x < y \leqslant z\}. \tag{2.1}$$

Then the *Plactic monoid* M_n is presented by $\langle A \mid \mathcal{R} \rangle$.

A *row* is a non-decreasing word in A^* (that is, a word $\alpha = \alpha_1 \cdots \alpha_k$, where $\alpha_i \in A$, in which $\alpha_i \leqslant \alpha_{i+1}$ for all $i = 1, \ldots, k-1$). Let $\alpha = \alpha_1 \cdots \alpha_k$ and $\beta = \beta_1 \cdots \beta_l$ (where $\alpha_i, \beta_i \in A$) be rows. The row α *dominates* the row β, denoted $\alpha \rhd \beta$, if $k \leqslant l$ and $\alpha_i > \beta_i$ for all $i = 1, \ldots, k$.

Any word $w \in A^*$ has a decomposition as a product of rows of maximal length $w = \alpha^{(1)} \cdots \alpha^{(k)}$. Such a word w is a *tableau* if $\alpha^{(i)} \rhd \alpha^{(i+1)}$ for all $i = 1, \ldots, k-1$. It is usual to write tableaux in a planar form, with the rows placed in order of domination and left-justified. For example, the tableau 6 3455 11235 is written as follows:

6				
3	4	5	5	
1	1	2	3	5

The set of tableaux form a cross-section of the Plactic monoid M_n [Lot02, Theorem 5.2.5]. For each $u \in A^*$, denote by $P(u)$ the unique tableau with $P(u) =_{M_n} u$. If u is a tableau, $P(u) = u$. Since the defining relations in the presentation $\langle A \mid \mathcal{R} \rangle$ preserve the number of symbols, it follows that $|P(u)| = |u|$ for all $u \in A^*$.

A *column* is a strictly decreasing word in A^* (that is, a word $\alpha = \alpha_k \cdots \alpha_1$, where $\alpha_i \in A$, in which $\alpha_{i+1} > \alpha_i$ for all $i = 1, \ldots, k-1$). [Notice the decreasing order of the subscripts on symbols of columns, so as to match the order of the symbols themselves.] This definition matches the notion of a column in the planar representation of a tableau.

Define a relation $\succeq$ on columns as follows: if $\alpha = \alpha_k \cdots \alpha_1$ and $\beta = \beta_l \cdots \beta_1$, then $\alpha \succeq \beta$ if and only if $k \geqslant l$ and $\alpha_i \leqslant \beta_i$ for all $i \leqslant l$. Thus $\alpha \succeq \beta$ if and only if the column α can appear immediately to the left of β in the planar representation of a tableau.

For any tableau w, denote by $C(w)$ the word obtained by reading (the planar representation of) that tableau column-wise from left to right and top to bottom. In the example above, $C(6\ 3455\ 11235) = 631\ 41\ 52\ 53\ 5$. Then $C(w) =_{M_n} w$ for all tableau w [Lot02, Problem 5.2.4].

The following result states the key combinatorial facts about tableaux:

THEOREM 2.1 ([Sch61, Theorems 1 & 2]; see also [Lot02, Theorem 5.1.1]). *Let* $u \in A^*$. *The number of columns in* $P(u)$ *is equal to the length of the longest non-decreasing subsequence in* u. *The number of rows in* $P(u)$ *is equal to the length of the longest decreasing subsequence in* u.

Let w be a tableau and let $\gamma \in A$. The unique tableau $P(w\gamma)$ equal to $w\gamma$ in M_n can be computed via Schensted's algorithm [Lot02, § 5.1–2], which we recall here:

ALGORITHM 2.2 (Schensted's algorithm).
   Input: A tableau w with rows $\alpha^{(1)}, \ldots, \alpha^{(k)}$ and a symbol $\gamma \in A$.
   Output: The unique tableau $P(w\gamma)$ equal to $w\gamma$ in M_n.
   Method:

1. If $\alpha^{(k)}\gamma$ is a row, the result is $\alpha^{(1)} \cdots \alpha^{(k)}\gamma$.

2. If $\alpha^{(k)}\gamma$ is not a row, then suppose $\alpha^{(k)} = \alpha_1 \cdots \alpha_l$ (where $\alpha_i \in A$) and let j be minimal such that $\alpha_j > \gamma$. Then the result is $P(\alpha^{(1)} \cdots \alpha^{(k-1)}\alpha_j)\alpha'^{(k)}$, where $\alpha'^{(k)} = \alpha_1 \cdots \alpha_{j-1}\gamma\alpha_{j+1} \cdots \alpha_l$.

Notice that in case 2, the algorithm replaces α_j by γ in the lowest row and recursively right-multiplies by α_j the tableau formed by all rows except the lowest. This is referred to as 'bumping' α_j to a higher row. When α_j is bumped, it will be inserted into the row above either in the same column or in some column further to the left, as shown in Figure 1. This happens because columns are strictly decreasing from top to bottom, so either the cell above α_j contains some symbol η greater than α_j, or α_j is the topmost element of its column. In the former case, α_j will be inserted so as to replace the leftmost symbol greater than α_j, which must either be to the left of η
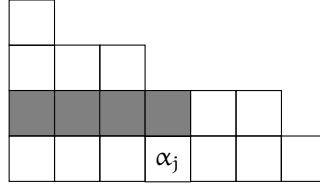
FIGURE 1. If the symbol α_j is bumped during Schensted's algorithm, it must be inserted into one of the shaded cells, since the cell above α_j must contain a symbol strictly greater than α_j.

or η itself, since rows are non-decreasing from left to right. In the latter case, α_j will be appended to the end of the row above and so will be placed either in the same column or further left.

For any word $u \in A^*$, the tableau $P(u)$ can be effectively computed by starting with the empty word, which is a valid tableau, and iteratively applying Schensted's algorithm.

2.2 *Biautomatic structures*

This subsection contains the definitions and basic results from the theory of automatic and biautomatic monoids needed hereafter. For further information on automatic semigroups, see [CRR01].

DEFINITION 2.3. Let A be an alphabet and let $\$$ be a new symbol not in A. Define the mapping $\delta_R : A^* \times A^* \to ((A \cup \{\$\}) \times (A \cup \{\$\}))^*$ by

$$(u_1 \cdots u_m, v_1 \cdots v_n) \mapsto \begin{cases} (u_1, v_1) \cdots (u_m, v_n) & \text{if } m = n, \\ (u_1, v_1) \cdots (u_n, v_n)(u_{n+1}, \$) \cdots (u_m, \$) & \text{if } m > n, \\ (u_1, v_1) \cdots (u_m, v_m)(\$, v_{m+1}) \cdots (\$, v_n) & \text{if } m < n, \end{cases}$$

and the mapping $\delta_L : A^* \times A^* \to ((A \cup \{\$\}) \times (A \cup \{\$\}))^*$ by

$$(u_1 \cdots u_m, v_1 \cdots v_n) \mapsto \begin{cases} (u_1, v_1) \cdots (u_m, v_n) & \text{if } m = n, \\ (u_1, \$) \cdots (u_{m-n}, \$)(u_{m-n+1}, v_1) \cdots (u_m, v_n) & \text{if } m > n, \\ (\$, v_1) \cdots (\$, v_{n-m})(u_1, v_{n-m+1}) \cdots (u_m, v_n) & \text{if } m < n, \end{cases}$$

where $u_i, v_i \in A$.

DEFINITION 2.4. Let M be a monoid. Let A be a finite alphabet representing a set of generators for M and let $L \subseteq A^*$ be a regular language such that every element of M has at least one representative in L. For each $a \in A \cup \{\varepsilon\}$, define the relations

$$L_a = \{(u, v) : u, v \in L, ua =_M v\}$$
$$_aL = \{(u, v) : u, v \in L, au =_M v\}.$$

The pair (A, L) is an *automatic structure* for M if $L_a\delta_R$ is a regular languages over $(A \cup \{\$\}) \times (A \cup \{\$\})$ for all $a \in A \cup \{\varepsilon\}$. A monoid M is *automatic* if it admits an automatic structure with respect to some generating set.

The pair (A, L) is a *biautomatic structure* for M if $L_a\delta_R$, $_aL\delta_R$, $L_a\delta_L$, and $_aL\delta_L$ are regular languages over $(A \cup \{\$\}) \times (A \cup \{\$\})$ for all $a \in A \cup \{\varepsilon\}$. A monoid M is *biautomatic* if it admits a biautomatic structure with respect to some generating set. [Note that biautomaticity implies automaticity.]

Unlike the situation for groups, biautomaticity for monoids and semigroups, like automaticity, is dependent on the choice of generating set [CRR01, Example 4.5]. However, for monoids, biautomaticity and automaticity are independent of the choice of *semigroup* generating sets [DRR99, Theorem 1.1].

Hoffmann & Thomas have made a careful study of biautomaticity for semigroups [HT05]. They distinguish four notions of biautomaticity for semigroups:

- *right-biautomaticity*, where $L_a\delta_R$ and $_aL\delta_R$ are regular languages;

- *left-biautomaticity*, where $L_a\delta_L$ and $_aL\delta_L$ are regular languages;

- *same-biautomaticity*, where $L_a\delta_R$ and $_aL\delta_L$ are regular languages;

- *cross-biautomaticity*, where $_aL\delta_R$ and $L_a\delta_L$ are regular languages.

These notions are all equivalent for groups and more generally for cancellative semi-groups [HT05, Theorem 1] but distinct for semigroups [HT05, Remark 1 & § 4]. In the sense used in this paper, 'biautomaticity' implies *all four* notions of biautomaticity above.

In proving certain that $R\delta_R$ or $R\delta_L$ is regular, where R is a relation on A^*, a useful strategy is to prove that R is a rational relation (that is, a relation recognized by a finite transducer [Ber79, Theorem 6.1]) and then apply the following result, which is a combination of [FS93, Corollary 2.5] and [HT05, Proposition 4]:

PROPOSITION 2.5. *If* $R \subseteq A^* \times A^*$ *is rational relation and there is a constant* k *such that* $\bigl||u| - |v|\bigr| \leqslant k$ *for all* $(u,v) \in R$, *then* $R\delta_R$ *and* $R\delta_L$ *are regular.*

REMARK 2.6. When constructing transducers to recognize particular relations, we will make use of certain strategies.

One strategy will be to consider a transducer reading elements of a relation R from *right to left*, instead of (as usual) left to right. In effect, such a transducer recognizes the reverse of R, which is the relation

$$R^{\mathrm{rev}} = \{(u^{\mathrm{rev}}, v^{\mathrm{rev}}) : (u,v) \in R\},$$

where u^{rev} and v^{rev} are the reverses of the words u and v respectively. Since the class of rational relations is closed under reversal [Ber79, p.65–66], constructing such a (right-to-left) transducer suffices to show that R is a rational relation.

Another important strategy will be for the transducer to non-deterministically guess some symbol yet to be read. More exactly, the transducer will non-deterministically select a symbol and store it in its state. When it later reads the relevant symbol, it checks it against the stored guessed symbol. If the guess was correct, the transducer continues. If the guess was wrong, the transducer enters a failure state. Similarly, the transducer can non-deterministically guess that it has reached the end of its input and enter an accept state. If it subsequently reads another symbol, it knows that its guess was wrong, and it enters a failure state.

3 COMPLETE REWRITING SYSTEM & GRÖBNER–SHIRSHOV BASIS

The aim of this section is to construct a finite complete rewriting system for M_n and so deduce the existence of a finite Gröbner–Shirshov basis for the the corresponding Plactic algebra.

The following lemma will play a crucial role in defining the rewriting system:

LEMMA 3.1. *Suppose* α *and* β *are columns with* $\alpha \not\succeq \beta$. *Then* $P(\alpha\beta)$ *contains at most two columns. Furthermore, if* $P(\alpha\beta)$ *contains exactly two columns, the left column contains more symbols than* α.

Proof of 3.1. Since α and β are strictly decreasing, the longest non-decreasing sequence in $\alpha\beta$ is at most 2, since it can contain at most one symbol from each of α and β. (It may have length 1 if every symbol in β is less than the minimum symbol in α.) Hence by Theorem 2.1, $P(\alpha\beta)$ contains at most two columns.

Suppose that $P(\alpha\beta)$ contains exactly two columns. Let $\alpha = \alpha_k \cdots \alpha_1$ and $\beta = \beta_l \cdots \beta_1$. Then since $\alpha \not\succeq \beta$, either $k < l$ or $\alpha_i > \beta_i$ for some $i \leqslant l$, as in the examples in Figure 2. In the first case, β is a decreasing subsequence of $\alpha\beta$ containing more symbols than α. In the second case, $\alpha_k \cdots \alpha_i \beta_i \cdots \beta_1$ is a decreasing subsequence of $\alpha\beta$ of length $k + 1$ and hence contains more symbols than α. In either case, $\alpha\beta$ contains a decreasing sequence of length greater than α, and so by Theorem 2.1, $P(\alpha\beta)$ contains more rows than there are symbols in α, and hence the left column of $P(\alpha\beta)$ contains more symbols than α. $\boxed{3.1}$
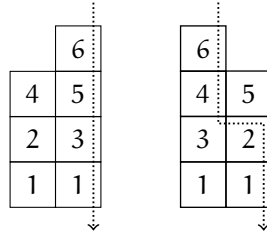
FIGURE 2. Two columns not related by $\succeq$ always contain a decreasing subsequence of length greater than the left column, as indicated by the dotted arrows.

To construct a finite complete rewriting system presenting M_n, introduce a new set of generators. Let

$$C = \{c_\alpha : \alpha \in A^+ \text{ is a column}\}$$

The idea is that each symbol c_α represents the element α of M_n. Thus the symbols $c_1, c_2, \ldots, c_n$ represent the original generating set for M_n, and so the set C also generates M_n. Furthermore, since the set of columns is finite (since a strictly decreasing sequence of elements of A has length at most $|A|$), the set C is finite. Notice that M_n is presented by $\langle C \mid \mathcal{R}' \cup \mathcal{S} \rangle$, where

$$\mathcal{R}' = \{(c_x c_z c_y, c_z c_x c_y) : x, y, z \in A \wedge x \leqslant y < z\}$$
$$\cup \{(c_y c_x c_z, c_y c_z c_x) : x, y, z \in A \wedge x < y \leqslant z\}$$
$$\mathcal{S} = \{(c_{\alpha_k \cdots \alpha_1}, c_{\alpha_k} \cdots c_{\alpha_1}) : \alpha_k \cdots \alpha_1 \text{ is a column}\};$$

the relations $\mathcal{R}'$ are simply those in $\mathcal{R}$ expressed using the symbols c_x (where $x \in A$), and those in $\mathcal{S}$ define the extra generators C_α where $|\alpha| \geqslant 2$.

Define a set of rewriting rules $\mathcal{T}$ on C^* as follows:

$$\mathcal{T} = \{c_\alpha c_\beta \to c_\gamma : \alpha \not\succeq \beta \wedge P(\alpha\beta) \text{ consists of one column } \gamma\} \tag{3.1}$$
$$\cup \{c_\alpha c_\beta \to c_\gamma c_\delta : \alpha \not\succeq \beta \wedge \tag{3.2}$$
$$P(\alpha\beta) \text{ consists of two columns, left col. } \gamma \text{ and right col. } \delta\}$$

Notice that every rule in $\mathcal{T}$ holds in the monoid M_n: this follows from the facts that $c_\zeta =_{M_n} \zeta$ for any column ζ, that $u =_{M_n} P(u)$ for all $u \in A^*$, and that $C(w) =_{M_n} w$ for all tableau w. For type (3.1) rules, $c_\alpha c_\beta =_{M_n} \alpha\beta =_{M_n} P(\alpha\beta) =_{M_n} C(P(\alpha\beta)) = \gamma =_{M_n} c_\gamma$; for type (3.2) rules, $c_\alpha c_\beta =_{M_n} \alpha\beta =_{M_n} P(\alpha\beta) =_{M_n} C(P(\alpha\beta)) = \gamma\delta =_{M_n} c_\gamma c_\delta$. Thus every rule in $\mathcal{T}$ is a consequence of the relations in $\mathcal{R}' \cup \mathcal{S}$.

Notice further that by $k - 1$ applications of type (3.1) rules, one can deduce every relation $(c_{\alpha_k \cdots \alpha_1}, c_{\alpha_k} \cdots c_{\alpha_1})$. Finally, it is easy to see that every relation in $\mathcal{R}'$ is also a consequence of those in $\mathcal{T}$. Thus M_n is presented by $\langle C \mid \mathcal{T} \rangle$. It remains to show that $(C, \mathcal{T})$ is a finite complete rewriting system.

By Lemma 3.1, if $\alpha \not\succeq \beta$, then $P(\alpha\beta)$ has at most two columns. Hence $\mathcal{T}$ contains a rewriting rule with left-hand side $c_\alpha c_\beta$ whenever $\alpha \not\succeq \beta$. Furthermore, since $P(\alpha\beta)$ is uniquely determined, $\mathcal{T}$ contains *exactly* one such rewriting rule, and hence the number of rules in $\mathcal{T}$ is finite.

LEMMA 3.2. *The rewriting system* $(C, \mathcal{T})$ *is noetherian.*

Proof of 3.2. Choose an ordering $\sqsubset$ on C that reverses the partial order induced by lengths of subscripts, in the sense that $c_\alpha \sqsubset c_\beta$ whenever $|\alpha| > |\beta|$. (Such an order must exist: simply reverse the order induces by the length of subscripts and then arbitrarily order elements with same-length subscripts.)

Let $\ll$ be the length-plus-lexicographic order on C^* induced by $\sqsubset$. That is:

$$c^{(1)} c^{(2)} \cdots c^{(k)} \ll d^{(1)} d^{(2)} \cdots d^{(l)}$$
$$\Longleftrightarrow k < l \vee \left(k = l \wedge (\exists i)(c^{(i)} \sqsubset d^{(i)} \wedge (\forall j < i)(c^{(j)} = d^{(j)})) \right),$$

where all symbols $c^{(h)}$ and $d^{(h)}$ lie in C. Then $\ll$ is a well-ordering of C^*. The aim is to prove that if $w \to w'$, then $w' \ll w$.

First, if the rule applied to obtain w' from w is of type (3.1), then $w = pc_\alpha c_\beta q$ and $w' = pc_\gamma q$ for some $p, q \in C^*$ and $c_\alpha, c_\beta, c_\gamma \in C$. So w' is a shorter word than w and so $w' \ll w$.

Second, if the rule applied to obtain w' from w is of type (3.2), then $w = pc_\alpha c_\beta q$ and $w' = pc_\gamma c_\delta q$ for some $p, q \in C^*$ and $c_\alpha, c_\beta, c_\gamma, c_\delta \in C$ with $P(\alpha\beta)$ having columns γ and δ. By Lemma 3.1, γ contains more symbols than α; that is, $|\gamma| > |\alpha|$. Hence, $c_\gamma \sqsubset c_\alpha$ by the choice of $\sqsubset$. So in the definition of $\ll$, we have $k = l$ and $c^{(i)} = c_\gamma \sqsubset c_\alpha = d^{(i)}$ and $c^{(j)} = d^{(j)}$ for all $j < i$ (where i is $|p| + 1$). Hence again $w' \ll w$.

Since $\ll$ is a well-ordering of C^*, there are no infinite $\ll$-infinite descending chains in C^*. Thus, since every application of a rule from $\mathcal{T}$ yields a $\ll$-preceding word, it follows that any sequence of rewriting using $\mathcal{T}$ must terminate. Hence $\mathcal{T}$ is noetherian. $\boxed{3.2}$

LEMMA 3.3. *The rewriting system* $(C, \mathcal{T})$ *is confluent.*

Proof of 3.3. Let $w \in C^*$. Since $(C, \mathcal{T})$ is noetherian by Lemma 3.2, applying $\mathcal{T}$ to w will always eventually yield an irreducible word. Let w' and w'' be irreducible words obtained from w. Suppose $w' = c_{\alpha^{(1)}} c_{\alpha^{(2)}} \cdots c_{\alpha^{(k)}}$. Now, since w' is irreducible, it does not contain any subword forming a left-hand side of a rule in $\mathcal{T}$. That is, there is no i such that $\alpha^{(i)} \not\succeq \alpha^{(i+1)}$. Equivalently, $\alpha^{(i)} \succeq \alpha^{(i+1)}$ for all i. Thus $\alpha^{(1)}\alpha^{(2)}\cdots\alpha^{(k)} = C(t')$ for some tableau t'. But t' must be the unique tableau with $t' =_{M_n} \alpha^{(1)}\alpha^{(2)}\cdots\alpha^{(k)} =_{M_n} w'$. Similarly, if $w'' = c_{\beta^{(1)}} c_{\beta^{(2)}} \cdots c_{\beta^{(l)}}$ then $\beta^{(1)}\beta^{(2)}\cdots\beta^{(l)} = C(t'')$, where t'' is the unique tableau with $t'' =_{M_n} \beta^{(1)}\beta^{(2)}\cdots\beta^{(l)} =_{M_n} w''$. But since $w =_{M_n} w' =_{M_n} w''$, and tableau form a cross-section of M_n, it follows that $t' = t''$. Hence $k = l$ and $\alpha^{(i)} = \beta^{(i)}$ for all $i = 1, \ldots, k$, and so $w' = w''$. Hence rewriting an arbitrary word $w \in C^*$ always terminates with a *unique* irreducible word. Thus the rewriting system $(C, \mathcal{T})$ is confluent. $\boxed{3.3}$

Lemmata 3.2 and 3.3, together with the finiteness of $\mathcal{T}$, yield the following result:

THEOREM 3.4. $(C, \mathcal{T})$ *is a finite complete rewriting system for the Plactic monoid* M_n.

The following corollary is immediate [SOK94]:

COROLLARY 3.5. *Every Plactic monoid has finite derivation type.*

By a result originally proved by Anick in different form [Ani86], but also proved by various other authors (see [Coh97]):

COROLLARY 3.6. *Every Plactic monoid is of type right and left* FP_∞.

Now let K be a field. Let $F = \{l - r : (l \to r) \in \mathcal{T}\} \subset K[C^*]$. Then the semigroup algebra $K[M_n]$ is isomorphic to the factor algebra $K[C^*]/\langle F \rangle$ (where $\langle F \rangle$ is the ideal generated by F) [Hey00, Proposition on p. 1]. Since $(C, \mathcal{T})$ is a finite complete rewriting system, F is a finite Gröbner–Shirshov basis for $K[M_n]$ [Hey00, Theorem on p. 1]. Furthermore, the order $\ll$ defined in the proof of Lemma 3.2 corresponds in $K[C^*]$ to the degree-lexicographic order. These remarks yield the following result:

THEOREM 3.7. *A Plactic algebra of arbitrary finite rank over an arbitrary field admits a finite Gröbner–Shirshov basis over* C *with respect to degree-lexicographic order.*

4 BIAUTOMATICITY

The aim of this section is to prove that the Plactic monoid M_n is biautomatic. We will prove biautomaticity with respect to the usual generating set A, but we will initially work with the generating set C. The first step is to define a language of representatives over C.

Let

$$K = \{c_{\alpha^{(1)}}c_{\alpha^{(2)}} \cdots c_{\alpha^{(k)}} : k \in \mathbb{N} \cup \{0\}, c_{\alpha^{(i)}} \in C, \alpha^{(j)} \succeq \alpha^{(j+1)} \text{ for all } j\}.$$

Notice that for any $c_{\alpha^{(1)}}, c_{\alpha^{(2)}}, \ldots, c_{\alpha^{(k)}} \in C$, we have $c_{\alpha^{(1)}}c_{\alpha^{(2)}} \cdots c_{\alpha^{(k)}} \in K$ if and only if $\alpha^{(1)} \alpha^{(2)} \cdots \alpha^{(k)}$ is the column reading of the corresponding tableau (that is, $\alpha^{(1)} \alpha^{(2)} \cdots \alpha^{(k)} = C(P(\alpha^{(1)}\alpha^{(2)} \cdots \alpha^{(k)}))$). Then K is a regular language over C, since an automaton need only store the previously-read symbol in its state in order to check that $\alpha^{(j)} \succeq \alpha^{(j+1)}$. Actually, K is the language of normal forms for the rewriting system $(C, \mathcal{T})$ [BO93, Lemma 2.1.3]. Duchamp & Krob [DK94, § 3.2] noted that this language K is a regular cross-section of the Plactic monoid, although their definition of K is rather different.

4.1 *Right-multiplication by transducer*

We will first of all prove that for any $\gamma \in A$ the relation K_{c_γ} is recognized by a finite transducer.

We imagine a transducer reading a pair of words

$$(c_{\alpha^{(1)}} \cdots c_{\alpha^{(k)}}, c_{\beta^{(1)}} \cdots c_{\beta^{(l)}}) \in K \times K$$

from right to left, with the aim of checking whether this pair is in K_{c_γ}. It is easiest to describe the transducer as reading symbols from the left tape and outputting symbols on the right tape. Essentially, the transducer will perform Schensted's algorithm using the alphabet C as a column representation of the tableau.

The transducer non-deterministically looks one symbol ahead (that is, further left) on the input tape. In its state, it stores a symbol η from A and a counter m which can take any value from $\{1, \ldots, n, \infty\}$. Initially, η is set to be γ and m is 1, corresponding to the bottom row of the tableau. The idea is that when $m \neq \infty$, the transducer is looking for the correct column in which to insert η in row m. Following Schensted's algorithm, the transducer will know if it has found the correct column $c_{\alpha^{(i)}}$ if the m-th symbol from the bottom of $\alpha^{(i)}$ is greater than γ and the m-th symbol from the bottom of $\alpha^{(i-1)}$ is less than or equal to γ. The crucial observation is that the transducer only needs a single right-to-left pass because when a symbol η is bumped, it is inserted into the next row either in the same column or in the some column further to the left, as was shown in Figure 1. When $m = \infty$, the transducer has completed the algorithm and simply reads symbols from the input tape and writes them on the output tape.

Initially, the transducer has $m = 1$, $\eta = \gamma$, and non-deterministically knows $c_{\alpha^{(k)}}$. If the bottom symbol of $\alpha^{(k)}$ is less than or equal to $\eta = \gamma$, then the transducer outputs c_γ before reading any input and then sets $m = \infty$.

When reading a symbol $c_{\alpha^{(i)}}$, the transducer non-deterministically knows $c_{\alpha^{(i-1)}}$ (or non-deterministically guesses that it has reached $c_{\alpha^{(1)}}$). As seen before, this is sufficient information to check whether the symbol η should be inserted into the column $\alpha^{(i)}$ at row m (bumping the m-th symbol from the bottom of $\alpha^{(i)}$). If such an insertion and bump is carried out, m is incremented by 1 and η replaced by the bumped symbol. The transducer may have to carry out several such insertions and bumps within the same column, but since there are only finitely many possibilities for $c_{\alpha^{(i)}}, c_{\alpha^{(i-1)}}, m$, and η, the result of carrying out all the necessary insertions and bumps can be stored in a finite lookup table. The transition function of the transducer can then be defined using this lookup table. Thus the transducer can calculate the value of the resulting column β and output c_β. If no such insertion and bumping is carried out, the transducer simply outputs $c_{\alpha^{(i)}}$.

Notice that when the transducer reads $c_{\alpha^{(i)}}$ and bumps symbols it may increment m to $|c_{\alpha^{(i)}}| + 1$. In this case, the transducer must insert η at the end of the m-th row, which corresponds to finding the first (rightmost) symbol $c_{\alpha^{(j)}}$ such that $|c_{\alpha^{(j-1)}}| \geq m$, and adding η to the top $\alpha^{(j)}$ to calculate the column β and output c_β. If the transducer reaches the leftmost end of the input word without finding such an $c_{\alpha^{(j)}}$, the m-th row is empty and so the symbol η is added to the top of $\alpha^{(1)}$. When a

symbol is added to the top of some $c_{\alpha(j)}$, the transducer has completed the algorithm and sets $m = \infty$.

Since it is recognized by a finite transducer, L_{c_γ} is a rational relation.

4.2 *Left-multiplication by transducer*

To prove that the relation $_{c_\gamma}K = \{(u, v) : u, v \in K, c_\gamma u =_{M_n} v\}$ is recognized by a finite transducer whenever $|\gamma| = 1$, we start with the following lemma, which is a straightforward consequence of Schensted's algorithm:

Lemma 4.1. *Let $\gamma \in A$ and let $\alpha = \alpha_p \cdots \alpha_1$ (where $\alpha_i \in A$) be a column. Then*

1. *$\gamma > \alpha_p$ if and only if $P(\gamma\alpha)$ is a single column $\gamma\alpha_p \cdots \alpha_1$.*

2. *r is minimal with $\gamma \leqslant \alpha_r$ if and only if $P(\gamma\alpha)$ has two columns: left column $\alpha_p \cdots \alpha_{r+1} \gamma \alpha_{r-1} \cdots \alpha_1$, and right column α_r.*

Notice that if $c_\gamma c_\alpha$ is reducible with respect to the rewriting system $(C, \mathcal{T})$, then $c_\gamma c_\alpha$ either rewrites to a single symbol $c_{\gamma\alpha}$ with $\gamma\alpha \succ \alpha$ or to a two-symbol word $c_{\alpha'} c_\eta$ with $\alpha' \succeq \alpha$ and $\gamma \leqslant \eta$.

Lemma 4.2. *Let $\alpha = \alpha_p \cdots \alpha_1$ and $\beta = \beta_q \cdots \beta_1$ be columns (where $\alpha_i, \beta_i \in A$) with $\alpha \succeq \beta$. Let $i \in \{1, \ldots, p\}$, and let η be the left-hand column of $P(\alpha_i\beta)$. Then $\alpha \succeq \eta$.*

Proof of 4.2. Since $\alpha \succeq \beta$, it follows that $p \geqslant q$ and $\alpha_j \leqslant \beta_j$ for all $j \leqslant q$. We distinguish two cases:

1. Suppose $P(\alpha_i\beta)$ has two columns. Then η has the form $\beta_q \cdots \beta_{r+1} \alpha_i \beta_{r-1} \cdots \beta_1$, where r is minimal with $\alpha_i \leqslant \beta_r$. So $|\eta| = |\beta|$ and thus $|\alpha| \geqslant |\eta|$. Notice that $r \leqslant i$, since otherwise we would have $\alpha_i \leqslant \beta_i < \beta_r$, contradicting the minimality of r. Therefore $\alpha_r \leqslant \alpha_i$ and for all $j \leqslant q$ with $j \neq r$ we have $\alpha_j \leqslant \beta_j$. Thus $\alpha \succeq \eta$.

2. Suppose $P(\alpha_i\beta)$ has one column (namely η). By Lemma 4.1 we have $\alpha_i > \beta_q$. Since $\beta_q \geqslant \alpha_q$ and α is a column, we conclude $i > q$. Thus $p > q$. Therefore η has the form $\alpha_i \beta_q \cdots \beta_1$ and $\alpha = \alpha_p \cdots \alpha_i \cdots \alpha_{q+1} \alpha_q \cdots \alpha_1$. Since $\alpha_{q+1} \leqslant \alpha_i$, it follows that $\alpha \succeq \eta$. $\boxed{4.2}$

Lemma 4.3. *Let $\gamma \in A$ and $c_\alpha, c_\beta \in C$ with $\alpha \succeq \beta$.*

1. *If $c_\gamma c_\alpha c_\beta \to c_{\alpha'} c_\eta c_\beta \to c_{\alpha'} c_{\beta'} c_\zeta$, then $\alpha' \succeq \beta'$.*

2. *If $c_\gamma c_\alpha c_\beta \to c_{\alpha'} c_\eta c_\beta \to c_{\alpha'} c_{\beta'}$, then $\alpha' \succeq \beta'$.*

3. *If $c_\gamma c_\alpha c_\beta \to c_{\alpha'} c_\beta$, then $\alpha' \succeq \beta$.*

Proof of 4.3. 1. From Lemma 4.1 and the remarks following it, we know that $\alpha' \succeq \alpha$ and η is a letter from α. Therefore, by Lemma 4.2, $\alpha \succeq \beta'$. Hence, since $\succeq$ is transitive, we have $\alpha' \succeq \beta'$.

2. The reasoning is the same as part 1.

3. From Lemma 4.1 and the remarks following it, we know that $\alpha' \succeq \alpha$. Since $\alpha \succeq \beta$ and $\succeq$ is transitive, we have $\alpha' \succeq \beta$. $\boxed{4.3}$

Let $c_{\alpha^{(1)}} \cdots c_{\alpha^{(k)}} \in K$. Recall that $\alpha^{(1)} \succeq \cdots \succeq \alpha^{(k)}$. Consider rewriting the word $c_\gamma c_{\alpha^{(1)}} \cdots c_{\alpha^{(k)}}$ to normal form using rules in $\mathcal{T}$. Suppose the rewriting proceeds as follows:

$$c_\gamma c_{\alpha^{(1)}} c_{\alpha^{(2)}} \cdots c_{\alpha^{(k)}}$$
$$\to c_{\alpha'^{(1)}} c_{\gamma_1} c_{\alpha^{(2)}} \cdots c_{\alpha^{(k)}}$$
$$\to \cdots$$
$$\to c_{\alpha'^{(1)}} c_{\alpha'^{(2)}} \cdots c_{\alpha'^{(i)}} c_{\gamma_i} c_{\alpha^{(i+1)}} \cdots c_{\alpha^{(k)}}.$$

By Lemma 4.3, $\alpha'^{(1)} \succeq \cdots \succeq \alpha'^{(i)}$. If $i = k$, then $\alpha'^{(i)} \succeq \gamma_i$ by the definition of $\mathcal{T}$. Suppose rewriting continues as follows:

$$\cdots$$
$$\to c_{\alpha'^{(1)}} \cdots c_{\alpha'^{(j)}} c_{\gamma_j} c_{\alpha^{(j+1)}} c_{\alpha^{(j+2)}} \cdots c_{\alpha^{(k)}}$$
$$\to c_{\alpha'^{(1)}} \cdots c_{\alpha'^{(j)}} c_{\alpha'^{(j+1)}} c_{\alpha^{(j+2)}} \cdots c_{\alpha^{(k)}}.$$

By Lemma 4.3, $c\alpha'^{(i)} \succeq \beta \succeq \alpha^{(j+2)}$. Hence rewriting $c_\gamma c_{\alpha^{(1)}} \cdots c_{\alpha^{(k)}}$ to normal form requires only a single left-to-right pass, which can be performed by a transducer: it simply stores the symbol c_{γ_i} in its state. Therefore the relation $_{c_\gamma} K$ can be recognized by a transducer.

4.3 *Deducing biautomaticity*

Let $\mathcal{Q} \subseteq C^* \times A^*$ be the relation

$$\left\{ (c_{\alpha^{(1)}} c_{\alpha^{(2)}} \cdots c_{\alpha^{(k)}}, \alpha^{(1)} \alpha^{(2)} \cdots \alpha^{(k)}) : k \in \mathbb{N} \cup \{0\}, \text{each } \alpha^{(i)} \text{ is a column} \right\}.$$

It is easy to see that $\mathcal{Q}$ is a rational relation. Let

$$L = K \circ \mathcal{Q} = \left\{ v \in A^* : (\exists u \in K)\big((u, v) \in \mathcal{Q} \big) \right\}.$$

Then L is a regular language over A that maps onto M_n, since the set of regular languages is closed under applying rational relations. (In fact, L is the set of column readings of tableaux, but this is not important for us.) Then for any $\gamma \in A$,

$$
\begin{aligned}
(u, v) \in L_\gamma &\iff u \in L \wedge v \in L \wedge u\gamma =_{M_n} v \\
&\iff (\exists u', v' \in K)((u, u') \in \mathcal{Q} \wedge (v, v') \in \mathcal{Q} \wedge u'c_\gamma =_{M_n} v') \\
&\iff (\exists u', v' \in K)((u, u') \in \mathcal{Q} \wedge (v, v') \in \mathcal{Q} \wedge (u', v') \in K_{c_\gamma}) \\
&\iff (u, v) \in \mathcal{Q}^{-1} \circ K_{c_\gamma} \circ \mathcal{Q}.
\end{aligned}
$$

Therefore, L_γ is a rational relation. Now, if $(u, v) \in L_\gamma$, then $|v| = |u| + 1$ since $u\gamma =_{M_n} v$ and the defining relations (2.1) preserve lengths of words. By Proposition 2.5, $L_\gamma \delta_R$ and $L_\gamma \delta_L$ are regular.

Similarly, from the fact that $_{c_\gamma} K$ is a rational relation, we deduce that $_\gamma L = \mathcal{Q}^{-1} \circ {}_{c_\gamma} K \circ \mathcal{Q}$ is rational and thus, by Proposition 2.5, that $_\gamma L \delta_R$ and $_\gamma L \delta_L$ are regular.

THEOREM 4.4. (A, L) *is a biautomatic structure for the Plactic monoid* M_n.

COROLLARY 4.5. *Let* B *be a generating set for the Plactic monoid* M_n. *Then* M_n *admits a biautomatic structure over* B.

Proof of 4.5. Since each generator in A admits no non-trivial decomposition in M_n, it follows that every element of A must also appear in B. Hence L is also a language over B. Let $b \in B$ and let $u_1 \cdots u_n \in A^*$ (where $u_i \in A$) be such that $b =_{M_n} u_1 \cdots u_n$. Then $L_b = L_{u_1} \circ L_{u_2} \circ \cdots \circ L_{u_n}$ and $_b L = {}_{u_1} L \circ {}_{u_2} L \circ \cdots \circ {}_{u_n} L$. So $L_b \delta_R$, $L_b \delta_L$, $_b L \delta_R$, and $_b L \delta_R$ are all regular (see, for example, [HT03, Proposition 2.4]). Hence (B, L) is a biautomatic structure for M_n. $\boxed{4.5}$

5 REFERENCES

[Ani86] D. J. Anick. 'On the homology of associative algebras'. *Trans. Amer. Math. Soc.*, 296, no. 2 (1986), pp. 641–659. DOI: 10.2307/2000383.

[Ber79] J. Berstel. *Transductions and context-free languages*, vol. 38 of *Leitfäden der Angewandten Mathematik und Mechanik [Guides to Applied Mathematics and Mechanics]*. B.G. Teubner, Stuttgart, 1979.

[BO93] R. V. Book & F. Otto. *String-Rewriting Systems*. Texts and Monographs in Computer Science. Springer-Verlag, New York, 1993.

[CHKT11] R. Corran, M. Hoffmann, D. Kuske, & R. M. Thomas. 'Singular Artin monoids of finite Coxeter type are automatic'. In *Language and Automata Theory and Applications*, vol. 6638 of *Lecture Notes in Computer Science*, pp. 250–261, 2011. DOI: 10.1007/978-3-642-21254-3_19.

[CJO10] F. Cedó, E. Jespers, & J. Okniński. 'Finitely presented algebras and groups defined by permutation relations'. *J. Pure Appl. Algebra*, 214, no. 7 (2010), pp. 1095–1102. DOI: 10.1016/j.jpaa.2009.09.015.

[CL11] Y. Chen & J. Li. 'New approach to Schensted–Knuth normal forms'. 2011. arXiv: 1106.4753v1.

[CO04] F. Cedó & J. Okniński. 'Plactic algebras'. *J. Algebra*, 274, no. 1 (2004), pp. 97–117. DOI: 10.1016/j.jalgebra.2003.12.004.

[Coh97] D. E. Cohen. 'String rewriting and homology of monoids'. *Math. Structures Comput. Sci.*, 7, no. 3 (1997), pp. 207–240. DOI: 10.1017/S0960129596002149.

[CQ08] Y. Chen & J. Qiu. 'Gröbner-Shirshov basis for the Chinese monoid'. *J. Algebra Appl.*, 7, no. 5 (2008), pp. 623–628. DOI: 10.1142/S0219498808003028.

[CRRT01] C. M. Campbell, E. F. Robertson, N. Ruškuc, & R. M. Thomas. 'Automatic semigroups'. *Theoret. Comput. Sci.*, 250, no. 1–2 (2001), pp. 365–391. DOI: 10.1016/S0304-3975(99)00151-6.

[DJM90] E. Date, M. Jimbo, & T. Miwa. 'Representations of $U_q(\mathfrak{gl}(n, \mathbf{C}))$ at $q = 0$ and the Robinson-Shensted [Schensted] correspondence'. In *Physics and mathematics of strings*, pp. 185–211. World Sci. Publ., Teaneck, NJ, 1990.

[DK94] G. Duchamp & D. Krob. 'Plactic-growth-like monoids'. In *Words, languages and combinatorics, II (Kyoto, 1992)*, pp. 124–142. World Sci. Publ., River Edge, NJ, 1994.

[DRR99] A. J. Duncan, E. F. Robertson, & N. Ruškuc. 'Automatic monoids and change of generators'. *Math. Proc. Cambridge Philos. Soc.*, 127, no. 3 (1999), pp. 403–409. DOI: 10.1017/S0305004199003722.

[ECH+92] D. B. A. Epstein, J. W. Cannon, D. F. Holt, S. V. F. Levy, M. S. Paterson, & W. P. Thurston. *Word Processing in Groups*. Jones & Bartlett, Boston, Mass., 1992.

[FS93] C. Frougny & J. Sakarovitch. 'Synchronized rational relations of finite and infinite words'. *Theoret. Comput. Sci.*, 108, no. 1 (1993), pp. 45–82. International Colloquium on Words, Languages and Combinatorics (Kyoto, 1990). DOI: 10.1016/0304-3975(93)90230-Q.

[Ful97] W. Fulton. *Young Tableaux*, vol. 35 of *London Mathematical Society Student Texts*. Cambridge University Press, Cambridge, 1997. With applications to representation theory and geometry.

[GK10] E. Güzel Karpuz. 'Complete rewriting system for the Chinese monoid'. *Appl. Math. Sci. (Ruse)*, 4, no. 21–24 (2010), pp. 1081–1087. URL: www.m-hikari.com/ams/ams-2010/ams-21-24-2010/karpuzAMS21-24-2010-2.pdf.

[Gre07] J. A. Green. *Polynomial representations of* GL_n, vol. 830 of *Lecture Notes in Mathematics*. Springer, Berlin, augmented edition, 2007. With an appendix on Schensted correspondence and Littelmann paths by K. Erdmann, Green and M. Schocker.

[Gro87] M. Gromov. 'Hyperbolic groups'. In S. M. Gersten, ed., *Essays in group theory*, vol. 8 of *Math. Sci. Res. Inst. Publ.*, pp. 75–263. Springer, New York, 1987.

[GS90] S. M. Gersten & H. B. Short. 'Small cancellation theory and automatic groups'. *Invent. Math.*, 102, no. 2 (1990), pp. 305–334. DOI: 10.1007/BF01233430.

[Hey00] A. Heyworth. 'Rewriting as a special case of non-commutative Gröbner basis theory'. 275 (2000), pp. 101–105. DOI: 10.1017/CBO9780511600609.009.

[HR12] D. F. Holt & S. Rees. 'Artin groups of large type are shortlex automatic with regular geodesics'. *Proc. Lond. Math. Soc. (3)*, 104, no. 3 (2012), pp. 486–512. DOI: 10.1112/plms/pdr035.

[HT03] M. Hoffmann & R. M. Thomas. 'Notions of automaticity in semigroups'. *Semigroup Forum*, 66, no. 3 (2003), pp. 337–367. DOI: 10.1007/s002330010161.

[HT05] M. Hoffmann & R. M. Thomas. 'Biautomatic semigroups'. In *Fundamentals of computation theory*, vol. 3623 of *Lecture Notes in Comput. Sci.*, pp. 56–67. Springer, Berlin, 2005. DOI: 10.1007/11537311_6.

[HU79]   J. E. Hopcroft & J. D. Ullman. *Introduction to Automata Theory, Languages, and Computation*. Addison–Wesley Publishing Co., Reading, Mass., 1979.

[Kas91]   M. Kashiwara.   'On crystal bases of the Q-analogue of universal enveloping algebras'.   *Duke Math. J.*, 63, no. 2 (1991), pp. 465–516.   DOI: 10.1215/S0012-7094-91-06321-0.

[Knu70]   D. E. Knuth. 'Permutations, matrices, and generalized Young tableaux'. *Pacific J. Math.*, 34 (1970), pp. 709–727. URL: projecteuclid.org/euclid.pjm/1102971948.

[KO14]   Ł. Kubat & J. Okniński. 'Gröbner-Shirshov Bases for Plactic Algebras'. *Algebra Colloq.*, 21, no. 4 (2014), pp. 591–596. DOI: 10.1142/S1005386714000534.

[KT97]   D. Krob & J. Y. Thibon. 'Noncommutative symmetric functions. IV. Quantum linear groups and Hecke algebras at q = 0'. *J. Algebraic Combin.*, 6, no. 4 (1997), pp. 339–376. DOI: 10.1023/A:1008673127310.

[Lit96]   P. Littelmann. 'A plactic algebra for semisimple Lie algebras'. *Adv. Math.*, 124, no. 2 (1996), pp. 312–331. DOI: 10.1006/aima.1996.0085.

[LLT95]   A. Lascoux, B. Leclerc, & J. Y. Thibon. 'Crystal graphs and q-analogues of weight multiplicities for the root system A_n'. *Lett. Math. Phys.*, 35, no. 4 (1995), pp. 359–374. DOI: 10.1007/BF00750843.

[Loh05]   M. Lohrey.   'Decidability and complexity in automatic monoids'.   *Internat. J. Found. Comput. Sci.*, 16, no. 4 (2005), pp. 707–722.   DOI: 10.1142/S0129054105003248.

[Lot02]   M. Lothaire. *Algebraic combinatorics on words*, vol. 90 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, Cambridge, 2002.

[LR34]   D. Littlewood & A. Richardson.   'Group characters and algebra'.   *Philos Trans. Roy. Soc. London Ser. A*, 233 (1934), pp. 99–141.   URL: http://www.jstor.org/stable/91293.

[LS78]   A. Lascoux & M. P. Schützenberger. 'Sur une conjecture de H. O. Foulkes'. *C. R. Acad. Sci. Paris Sér. A-B*, 286, no. 7 (1978), pp. A323–A324.

[LS81]   A. Lascoux & M. P. Schützenberger. 'Le monoïde plaxique'. In *Noncommutative structures in algebra and geometric combinatorics (Naples, 1978)*, vol. 109 of *Quad. "Ricerca Sci."*, pp. 129–156. CNR, Rome, 1981.

[LS90]   A. Lascoux & M. P. Schützenberger. 'Keys & standard bases'. In *Invariant theory and tableaux (Minneapolis, MN, 1988)*, vol. 19 of *IMA Vol. Math. Appl.*, pp. 125–144. Springer, New York, 1990.

[LT96]   B. Leclerc & J. Y. Thibon.   'The Robinson-Schensted correspondence, crystal bases, and the quantum straightening at q = 0'. *Electron. J. Combin.*, 3, no. 2 (1996), pp. Research Paper 11, approx. 24 pp. (electronic). The Foata Festschrift. URL: http://www.combinatorics.org/Volume_3/Abstracts/v3i2r11.html.

[OSKM98]   F. Otto, A. Sattler-Klein, & K. Madlener. 'Automatic monoids versus monoids with finite convergent presentations'. In *Rewriting techniques and applications (Tsukuba, 1998)*, vol. 1379 of *Lecture Notes in Comput. Sci.*, pp. 32–46. Springer, Berlin, 1998. DOI: 10.1007/BFb0052359.

[Ott00]   F. Otto. 'On Dehn functions of finitely presented bi-automatic monoids'. *J. Autom. Lang. Comb.*, 5, no. 4 (2000), pp. 405–419.

[Pic06]   M. Picantin. 'Finite transducers for divisibility monoids'. *Theoret. Comput. Sci.*, 362, no. 1-3 (2006), pp. 207–221. DOI: 10.1016/j.tcs.2006.06.019.

[Sch61]   C. Schensted. 'Longest increasing and decreasing subsequences'. *Canad. J. Math.*, 13 (1961), pp. 179–191. DOI: 10.4153/CJM-1961-015-3.

[Sch97]   M. P. Schützenberger. 'Pour le monoïde plaxique'. *Math. Inform. Sci. Humaines*, , no. 140 (1997), pp. 5–10. URL: www.ehess.fr/revue-msh/pdf/N140R764.pdf.

[Ser10]   L. Serrano. 'The shifted plactic monoid'. *Math. Z.*, 266, no. 2 (2010), pp. 363–392. DOI: 10.1007/s00209-009-0573-0.

[SOK94]   C. C. Squier, F. Otto, & Y. Kobayashi.   'A finiteness condition for rewriting systems'.   *Theoret. Comput. Sci.*, 131, no. 2 (1994), pp. 271–294.   DOI: 10.1016/0304-3975(94)90175-9.

[Ufn98] V. Ufnarovski. 'Introduction to noncommutative Gröbner bases theory'. In *Gröbner bases and applications (Linz, 1998)*, vol. 251 of *London Math. Soc. Lecture Note Ser.*, pp. 259–280. Cambridge Univ. Press, Cambridge, 1998. DOI: 10.1017/CBO9780511565847.015.