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

Robert Gray
(joint work with A. J. Cain and A. Malheiro)

Geometric, Combinatorial & Dynamics Aspects of Semigroups and Groups

On the occasion of the 60th birthday of Stuart Margolis
Bar-Ilan, Israel, June 2013
A tableau

<table>
<thead>
<tr>
<th>6</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
A tableau

![Tableau diagram]

Properties

- Rows read left-to-right are non-decreasing.
- Columns read down are strictly decreasing.
- Never have a longer row above a strictly shorter one.
Outline

Plactic monoid and algebras
  Tableaux and the Schensted insertion algorithm
  The Plactic monoid

Rewriting systems
  Finite complete rewriting systems for Plactic monoids
  Gröbner–Shirshov bases for Plactic algebras

Automaticity
  Biautomatic structures for Plactic monoids

Related results and future work
Tableaux

Let $n \in \mathbb{N}$, and let $A$ be the finite ordered alphabet

$$A = \{1 < 2 < \cdots < n\}.$$  

Definitions

**Row** a non-decreasing word $w \in A^*$ (e.g. 11224556)

**Domination** The row $\alpha = \alpha_1 \cdots \alpha_k$ dominates the row $\beta = \beta_1 \cdots \beta_l$, denoted $\alpha \triangleright \beta$, if $k \leq l$ and $\alpha_i > \beta_i$ for all $i \leq k$.

i.e.

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha_1$</td>
<td>$\alpha_2$</td>
<td>$\alpha_3$</td>
<td>$\alpha_4$</td>
<td></td>
</tr>
<tr>
<td>$\triangleright$</td>
<td>$\triangleright$</td>
<td>$\triangleright$</td>
<td>$\triangleright$</td>
<td>$\triangleright$</td>
</tr>
<tr>
<td>$\beta_1$</td>
<td>$\beta_2$</td>
<td>$\beta_3$</td>
<td>$\beta_4$</td>
<td>$\beta_5$</td>
</tr>
</tbody>
</table>

▶ We write tableaux in a planar form with rows placed in order of domination and left-justified.
Tableaux

Let \( n \in \mathbb{N} \), and let \( A \) be the finite ordered alphabet

\[
A = \{ 1 < 2 < \cdots < n \}.
\]

Definitions

**Row**  a non-decreasing word \( w \in A^* \) (e.g. 111224556)

**Domination** The row \( \alpha = \alpha_1 \cdots \alpha_k \) dominates the row \( \beta = \beta_1 \cdots \beta_l \), denoted \( \alpha \triangleright \beta \), if \( k \leq l \) and \( \alpha_i > \beta_i \) for all \( i \leq k \).

\[
\begin{array}{cccccc}
\alpha_1 & \alpha_2 & \alpha_3 & \alpha_4 \\
\lor & \lor & \lor & \lor \\
\beta_1 & \beta_2 & \beta_3 & \beta_4 & \beta_5 & \beta_6.
\end{array}
\]

**Tableau** Any word \( w \in A^* \) has a decomposition as a product of rows of maximal length \( w = \alpha^{(1)} \cdots \alpha^{(k)} \).

Then \( w \) is a **tableau** if \( \alpha^{(i)} \triangleright \alpha^{(i+1)} \) for all \( i \).

- We write tableaux in a planar form with rows placed in order of domination and left-justified.
Tableaux - in pictures

Example
Let $A = \{1 < 2 < 3 < 4 < 5\}$, and consider $\alpha = 325114 \in A^*$
$\alpha = 325114 = 3 \ 25 \ 114 = \alpha^{(1)} \alpha^{(2)} \alpha^{(3)}$

\[
\begin{array}{ccc}
3 & & \\
2 & 5 & \\
1 & 1 & 4 \\
\end{array}
\]

- Columns read down are strictly decreasing.
- Never have a longer row above a strictly shorter one.
- Conclusion: $\alpha$ is a tableau.

Notes:
- Symbols in tableaux are allowed to repeat.
- Rows can be arbitrarily long while columns have height bounded by $n$.
- There are infinitely many tableaux over $A = \{1 < \cdots < n\}$. 
Schensted’s algorithm - Easier done than said

- Associates to each word \( w \in A^* \) a tableau \( t = P(w) \).
- The algorithm which produces \( P(w) \) is recursive.
- \( P(w) \) is obtained by permuting the symbols of \( w \) in a certain way.

**Input:** A tableau \( w \) with rows \( \alpha^{(1)}, \ldots, \alpha^{(k)} \) and a symbol \( \gamma \in A \).

**Output:** The tableau \( P(w\gamma) \).
Schensted’s algorithm - Easier done than said

- Associates to each word $w \in A^*$ a tableau $t = P(w)$.
- The algorithm which produces $P(w)$ is recursive.
- $P(w)$ is obtained by permuting the symbols of $w$ in a certain way.

**Input:** A tableau $w$ with rows $\alpha^{(1)}, \ldots, \alpha^{(k)}$ and a symbol $\gamma \in A$.

**Output:** The tableau $P(w\gamma)$.

**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 results is:

$$P(\alpha^{(1)} \cdots \alpha^{(k-1)} \alpha_j) \alpha'^{(k)},$$

where $\alpha'^{(k)} = \alpha_1 \cdots \alpha_{j-1} \gamma \alpha_{j+1} \cdots \alpha_l$.

**Bumping**

In case 2, the algorithm replaces $\alpha_j$ by $\gamma$ in the lowest row and recursively right-multiplies by $\alpha_j$ the tableau formed by all rows except the lowest.
Schensted’s algorithm example

\[ n = 5, \quad \alpha = 132541, \quad \text{compute } P(\alpha) \]
Schensted’s algorithm example

\[ n = 5, \quad \alpha = 132541, \quad \text{compute } P(\alpha) \]
Schensted’s algorithm example

\[ n = 5, \quad \alpha = 132541, \quad \text{compute } P(\alpha) \]
Schensted’s algorithm example

\( n = 5, \quad \alpha = 132541, \quad \text{compute } P(\alpha) \)
Schensted’s algorithm example

\[ n = 5, \quad \alpha = 132541, \quad \text{compute } P(\alpha) \]
Schensted’s algorithm example

\[ n = 5, \quad \alpha = 132541, \quad \text{compute } P(\alpha) \]
Schensted’s algorithm example

\[ n = 5, \quad \alpha = 132541, \quad \text{compute } P(\alpha) \]
Schensted’s algorithm example

\[ n = 5, \quad \alpha = 132541, \quad \text{compute } P(\alpha) \]
Schensted’s algorithm example

\[ n = 5, \quad \alpha = 132541, \quad \text{compute } P(\alpha) \]
Schensted’s algorithm example

\[ n = 5, \quad \alpha = 132541, \quad \text{compute } P(\alpha) \]
Schensted’s algorithm example

\[ n = 5, \quad \alpha = 132541, \quad \text{compute } P(\alpha) \]
Schensted’s algorithm example

\[ n = 5, \quad \alpha = 132541, \quad \text{compute } P(\alpha) \]
Schensted’s algorithm example

\[ n = 5, \quad \alpha = 132541, \quad \text{compute } P(\alpha) \]
Schensted’s algorithm example

\[ n = 5, \quad \alpha = 132541, \quad \text{compute } P(\alpha) \]
Schensted’s algorithm example

\[ n = 5, \quad \alpha = 132541, \quad \text{compute } P(\alpha) \]
Schensted’s algorithm example

\[ n = 5, \quad \alpha = 132541, \quad \text{compute } P(\alpha) \]

\[
\begin{array}{ccc}
3 & & \\
1 & 2 & 4 \\
\end{array}
\]
Schensted’s algorithm example

\[ n = 5, \quad \alpha = 132541, \quad \text{compute } P(\alpha) \]
Schensted’s algorithm example

\[ n = 5, \quad \alpha = 132541, \quad \text{compute } P(\alpha) \]

\[
\begin{array}{ccc}
3 & 5 \\
1 & 2 & 4 \\
\end{array}
\]
Schensted’s algorithm example

\[ n = 5, \quad \alpha = 132541, \quad \text{compute } P(\alpha) \]

\begin{verbatim}
3 5
1 1 4
\end{verbatim}

`Bumped'
Schensted’s algorithm example

\( n = 5, \quad \alpha = 132541, \quad \text{compute} \ P(\alpha) \)
Schensted’s algorithm example

\[ n = 5, \quad \alpha = 132541, \quad \text{compute } P(\alpha) \]
Schensted’s algorithm example

\[ n = 5, \quad \alpha = 132541, \quad \text{compute } P(\alpha) \]
Schensted’s algorithm example

\[ n = 5, \quad \alpha = 132541, \quad \text{compute } P(\alpha) \]
Schensted’s algorithm example

\[ n = 5, \quad \alpha = 132541, \quad \text{compute } P(\alpha) \]
Schensted’s algorithm example

\[ n = 5, \quad \alpha = 132541, \quad \text{compute } P(\alpha) \]
Schensted’s algorithm example

\[ n = 5, \quad \alpha = 132541, \quad \text{compute } P(\alpha) \]

\[
\begin{array}{ccc}
3 & & \\
2 & 5 & \\
1 & 1 & 4 \\
\end{array}
\]

Conclusion

\[ P(\alpha) = P(132541) = 3\ 25\ 114 = 325114, \text{ which is a tableau.} \]

Fact

If \( w \in A^* \) is already a tableau then \( P(w) = w \) in \( A^* \).

e.g. \( P(325114) = 325114 \).
The Plactic monoid

\[ A = \{1 < 2 < \cdots < n\} \]

Define an equivalence relation \( \sim \) on \( A^* \) by

\[ u \sim v \iff P(u) = P(v) \text{ in } A^*. \]

Theorem (Knuth (1970))

The equivalence relation \( \sim \) is a congruence on the free monoid \( A^* \).
The quotient \( M_n = A^*/\sim \) is called the Plactic monoid of rank \( n \).

So, the Plactic monoid \( M_n \) is the monoid of tableaux:

Elements  The set of all tableaux over \( A = \{1 < 2 < \cdots < n\} \).

Multiplication  Given tableaux \( u \) and \( v \), their product is \( u \cdot v = P(uv) \).

Example

\[
\begin{array}{ccc}
5 & & \\
2 & 4 & \\
\end{array}
\cdot
\begin{array}{ccc}
4 & & \\
1 & 3 & \\
\end{array} =
\begin{array}{ccc}
5 & & \\
2 & 4 & \\
1 & 3 & 4 \\
\end{array}
\]
A finite presentation for the Plactic monoid $M_n$

- For words $u, v$ of length $\leq 2$ we have $u \sim v \iff u \equiv w$.
- The Knuth relations $= \{ \text{all relations } u \sim v \text{ for words of length } 3 \}$.
- In fact, these relations alone are enough to define the monoid.

**Theorem (Knuth (1970))**

Let $n \in \mathbb{N}$. Let $A$ be the finite ordered alphabet $\{1 < 2 < \ldots < n\}$.

Let $R$ be the set of defining relations:

$$zxy = xzy \quad x \leq y < z,$$

$$yzx = yxz \quad x < y \leq z.$$

Then the Plactic monoid $M_n$ is finitely presented by $\langle A|R \rangle$. 
The Plactic monoid

- Has origins in work of Schensted (1961) and Knuth (1970) concerned with combinatorial problems on Young tableaux.

Due to close relations to Young tableaux, has become a tool in several aspects of representation theory and algebraic combinatorics.

Applications of the Plactic monoid

- proof of the Littlewood–Richardson rule for Schur functions (an important result in the theory of symmetric functions);
  - see appendix of J. A. Green’s “Polynomial representations of GL_n”.
- a combinatorial description of the Kostka–Foulkes polynomials, which arise as entries of the character table of the finite linear groups.

M. P. Schützenberger ‘Pour le monoïde plaxique’ (1997)

Argues that the Plactic monoid ought to be considered as “one of the most fundamental monoids in algebra”.
Complete rewriting systems

$X$ - alphabet, $R \subseteq X^* \times X^*$ - rewrite rules, $\langle X \mid R \rangle$ - rewriting system

Write $r = (r_+, r_-) \in R$ as $r_+ \rightarrow r_-.$

Define a binary relation $\rightarrow_R$ on $X^*$ by

\[ u \rightarrow_R v \iff u \equiv w_1 r_+ w_2 \text{ and } v \equiv w_1 r_- w_2 \]

for some $(r_+, r_-) \in R$ and $w_1, w_2 \in X^*.$

$\rightarrow_R^*$ is the transitive and reflexive closure of $\rightarrow_R.$

**Noetherian:** No infinite descending chain

\[ w_1 \rightarrow_R w_2 \rightarrow_R \cdots \rightarrow_R w_n \rightarrow_R \cdots \]

**Confluent:** Whenever $u \rightarrow_R^* v$ and $u \rightarrow_R^* v'$

there is a word $w \in X^*$:

\[ v \rightarrow_R^* w \text{ and } v' \rightarrow_R^* w \]

**Definition:** $R$ is complete if it is both noetherian & confluent.
Complete rewriting systems

- alphabet, \( R \subseteq X^* \times X^* \) - rewrite rules

Let \( M = X^*/\leftrightarrow^*_R \) be the monoid defined by \( \langle X \mid R \rangle \) where \( \leftrightarrow^*_R \) is the congruence generated by \( R \).

A word \( u \) is irreducible if no reduction \( u \rightarrow^*_R v \) can be applied.

- If \( R \) is a noetherian rewriting system, each congruence class of \( M = X^*/\leftrightarrow^*_R \) contains at least one irreducible element.

**Proposition**

Assuming \( R \) is noetherian, then \( R \) is a complete rewriting system \( \iff \) each congruence class of \( M = X^*/\leftrightarrow^*_R \) contains exactly one irreducible word.

- \( \langle X \mid R \rangle \) is a finite complete rewriting system if it is complete (noetherian and confluent) and \( |X| < \infty \) and \( |R| < \infty \).
Finite complete rewriting systems for the Plactic monoid

Kubat and Okniński (2010) showed...

Let $A = \{1 < 2 < 3\}$. The eight Knuth relations

$$zxy \rightarrow xzy \ (x \leq y < z), \quad yzx \rightarrow yxz \ (x < y \leq z) \quad x, y, z \in A,$$

taken together with the following rewrite rules:

$$3212 \rightarrow 2321, \quad 32131 \rightarrow 31321, \quad 32321 \rightarrow 32132,$$

gives a finite complete rewriting system defining $M_3$. 
Kubat and Okniński (2010) showed...

- Let $A = \{1 < 2 < 3\}$. The eight Knuth relations

\[ zxy \rightarrow xzy \ (x \leq y < z), \quad yzx \rightarrow yxz \ (x < y \leq z) \quad x, y, z \in A, \]

taken together with the following rewrite rules:

\[ 3212 \rightarrow 2321, \quad 32131 \rightarrow 31321, \quad 32321 \rightarrow 32132, \]

gives a finite complete rewriting system defining $M_3$.

- Their results show that for higher ranks the same approach does not yield a finite complete rewriting system i.e. for $n \geq 4$, starting with:

\[ zxy \rightarrow xzy \ (x \leq y < z), \quad yzx \rightarrow yxz \ (x < y \leq z) \quad x, y, z \in A, \]

then there is no finite set of rules $u \rightarrow v$ (with $v <_{\text{lex}} u$) holding in $M_n$, that can be added to obtain a complete rewriting system defining $M_n$. 
This leaves the question...

**Question**

Does the Plactic monoid $M_n$ admit a presentation by a finite complete rewriting system (with respect to some finite generating set)?
Change of viewpoint

\[ A = \{1 < 2 < \cdots < n\} \]

**Column** a strictly decreasing word in \( A^* \) (e.g. 98532)

**Note:** There are only finitely many columns (since height bounded by \( n \)).

**Column readings**

Denote by \( C(w) \) (with \( w \) a tableau) the word obtained by reading that tableau column-wise from left to right and top to bottom.

**Exercise:** \( C(w) = w \) in \( M_n \) for any tableau \( w \).

**Example**

We have the tableau
\[ w = 3 \ 25 \ 114 = 325114, \text{ with } C(w) = 321 \ 51 \ 4 = 321514, \text{ and } \]
\[ 325114 = 321514 \text{ in } M_5. \]
Working with columns

Thus, the set of column readings of the tableaux gives an alternative set of normal forms in $A^*$ for the elements of $M_n$.

Define a new alphabet representing the set of all columns:

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

Column readings give a canonical way of expressing each element (tableau) of $M_n$ uniquely as a product of the generators $C$.

The idea

Seek a complete rewriting system for the Plactic monoid with respect to $C$. 
Working with columns

Thus, the set of column readings of the tableaux gives an alternative set of normal forms in $A^*$ for the elements of $M_n$.

Define a new alphabet representing the set of all columns:

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

Column readings give a canonical way of expressing each element (tableau) of $M_n$ uniquely as a product of the generators $C$.

The idea
Seek a complete rewriting system for the Plactic monoid with respect to $C$.

Compatible columns: Define a relation $\succeq$ on columns as follows: if $\alpha = \alpha_k \cdots \alpha_1$ and $\beta = \beta_l \cdots \beta_1$ are columns,

$$\alpha \succeq \beta \iff k \geq l \text{ and } \alpha_i \leq \beta_i \text{ for all } i \leq l.$$ 

Thus $\alpha \succeq \beta$ if and only if the column $\alpha$ can appear immediately to the left of $\beta$ in the planar representation of a tableau.
Multiplying pairs of columns

Compatible columns: Product $P(uw)$ where $u \succeq w$.

Does not give rise to a relation between words over $C^*$.

Incompatible columns: Symbols in $w$ all strictly less than those in $u$.

Then $P(uw)$ has a single column.
Multiplying pairs of columns

Incompatible columns: Left column shorter than right.

Incompatible columns: A strict increase in one of the rows.

Note: In both of these examples (1) the product again has two columns \( w \) and \( x \), and (2) the left column \( w \) of the product is strictly taller than the left column \( u \) of the original pair \( u, v \) of columns.
Multiplying pairs of columns

This is true in general:

**Key Lemma**
Suppose $\alpha$ and $\beta$ are columns with $\alpha \nleq \beta$. 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 $\alpha$.

This result is proved by applying the following classical result:

**Theorem (Schensted (1961))**
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 $u$. 
Finite complete rewriting system for Plactic monoids

\[ C = \{ c_\alpha : \alpha \in A^* \text{ is a column} \} \]

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

\[ \mathcal{T} = \{ c_\alpha c_\beta \rightarrow c_\gamma : \alpha \not\preceq \beta \land P(\alpha\beta) \text{ consists of one column } \gamma \} \]

\[ \cup \{ c_\alpha c_\beta \rightarrow c_\gamma c_\delta : \alpha \not\preceq \beta \land \]

\[ P(\alpha\beta) \text{ consists of two columns, left col. } \gamma \text{ and right col. } \delta \} \]

**Lemma**

The Plactic monoid \( M_n \) is finitely presented by \( \langle C \mid \mathcal{T} \rangle \).

We claim that \( \langle C \mid \mathcal{T} \rangle \) is a finite complete rewriting system.
\((C, \mathcal{T})\) is noetherian

- ordering on \(C\) such that \(c_\alpha \sqsubset c_\beta\) whenever \(|\alpha| > |\beta|\);
- \(\ll\) – the length-plus-lexicographic order on \(C^*\) induced by \(\sqsubset\) (which is a well-ordering of \(C^*\));
- Applying the key lemma: If \(w \rightarrow_\gamma w'\) then \(w' \ll w\).
$(C, \mathcal{T})$ is confluent

- Let $w \in C^*$ be arbitrary.
- Noetherian $\Rightarrow$ applying $\mathcal{T}$ to $w$ will eventually yield some irreducible

$$w' \equiv c_1 c_2 \ldots c_k \in C^*.$$  

- $w'$ irreducible $\Rightarrow c_i \succeq c_{i+1}$ for all $i$.
- Thus the columns $c_1, c_2, \ldots, c_k$ form a tableau which is precisely the element of the Plactic monoid $M_n$ represented by the word $w \in C^*$.
- Thus $w'$ is uniquely determined by $w$ i.e. each $w \in C^*$ reduces to a unique irreducible word under $\rightarrow_{\mathcal{T}}$. 

Diagram:

```
C1  C2
   |
   v
C3  C4
    |
    v
C5  C6  C7
```
Finite complete rewriting system for Plactic monoids

Theorem (Cain, RG, Malheiro (2012))

\((C, T)\) is a finite complete rewriting system for the Plactic monoid \(M_n\).

Note

Chen and Li (2011) exhibit an infinite complete rewriting systems for Plactic monoids over the (infinite) set of rows of tableaux.
Plactic algebras

$K$ - a field, $K[M_n]$ - the Plactic algebra of rank $n$ over $K$

Various aspects of Plactic algebras have been considered:

- Cedó, Okniński (2004): structure of Plactic algebras of ranks 2 and 3 (investigated properties: semiprimitive, semiprime, and prime);
- Kubat, Okniński (2012): Plactic algebra of rank 3 studied (including description of minimal prime ideals);

Are important special cases in general study of algebras defined by homogeneous semigroup relations, including

- Chinese algebras;
- algebras defined by permutation relations;
- algebras related to the quantum Yang–Baxter equation.

See work of Cedó, Jaszuńska, Jespers, Kubat, Okniński, and others...
Complete rewriting systems and Gröbner–Shirshov bases

$K$ - field, $\langle A, R \rangle$ - finite rewriting system defining a monoid $M$
$K[M]$ - corresponding semigroup algebra

Let $F = \{l - r : (l \to r) \in R\} \subset K[A^*]$.

**Proposition.** The semigroup algebra $K[M]$ is isomorphic to the factor algebra $K[A^*]/\langle F \rangle$, where $\langle F \rangle$ is the ideal generated by $F$.

**Proposition.** If $\langle A, R \rangle$ is a finite complete rewriting system then

\[ F = \{l - r : (l \to r) \in R\} \subset K[A^*] \]

is a finite Gröbner–Shirshov basis for $K[M] \cong K[A^*]/\langle F \rangle$.

Heyworth (1999) – gives a ‘dictionary’ linking these two worlds:

complete rewrite system $\leftrightarrow$ Gröbner–Shirshov basis
Knuth–Bendix completion algorithm $\leftrightarrow$ Buchberger algorithm
The results on finite complete rewriting systems proved by Kubat and Okniński were actually expressed these terms:

**Theorem (Kubat and Okniński (2010))**

Let $K[M_n]$ be the Plactic algebra of rank $n$ over a field $K$.

1. If $n = 3$ then $K[M_n]$ has a finite Gröbner–Shirshov basis.
2. If $n > 3$ then every Gröbner–Shirshov basis of $K[M_n]$ (associated to the degree-lexicographic ordering on $A$) is infinite.

Our result may also be expressed in these terms:

**Theorem (Cain, RG, Malheiro (2012))**

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.
Automatic structures

Automatic groups and monoids

- Automatic groups
  - Capture a large class of groups with easily solvable word problem
  - Examples: finite groups, free groups, free abelian groups, various small cancellation groups, Artin groups of finite and large type, Braid groups, hyperbolic groups.

- Automatic semigroups and monoids
  - Classes of monoids that have been shown to be automatic include divisibility monoids and singular Artin monoids of finite type.

**Defining property:** existence of rational set of normal forms (with respect to some finite generating set $A$) such that $\forall a \in A$, there is a finite automaton recognising pairs of normal forms that differ by multiplication by $a$.

**Proposition (Campbell et al. (2001))**
Automatic monoids have word problem solvable in quadratic time.
Plactic monoids and automaticity

1. Plactic monoids have word problem solvable in quadratic time
   ▶ a consequence of the Schensted insertion algorithm
2. Automatic monoids have word problem solvable in quadratic time

These two facts led Efim Zelmanov during the conference

*Groups and Semigroups: Interactions and Computations* (Lisbon, July 2011)

to ask the following natural question:

“Are Plactic monoids automatic?”
Plactic monoids are biautomatic

$A = \{1 < 2 < \cdots < n\}, \quad M_n$ - Plactic monoid of rank $n$

$L = \text{the set of all column readings of tableaux.}$

$L \subseteq A^*$ is a regular language over $A$ that maps onto $M_n$.

Theorem (Cain, RG, Malheiro (2012))

$(A, L)$ is a biautomatic structure for the Plactic monoid $M_n$.

- Biautomatic = the strongest form of automaticity for monoids.
- Beginning with the finite complete rewriting system obtained above, we show how for Plactic monoids finite transducers may be constructed to perform left (respectively right) multiplication by a generator.

Corollary (Cain, RG, Malheiro (2012))

Let $B$ be a finite generating set for the Plactic monoid $M_n$. Then $M_n$ admits a biautomatic structure over $B$. 

Related results and future work

- The Chinese monoid $C_n$
  - $A = \{1 < 2 < \ldots < n\}$, relations $\{(zyx, zyx), (zxy, yzx) : x \leq y \leq z\}$.
  - Using Chinese staircase representation of Cassaige et al. (2001) we prove

Theorem (Cain, RG, Malheiro (2013)) Chinese monoids are biautomatic.

- Monoids defined by multihomogeneous presentations
  - We have examples of multihomogeneous presentations that:
    - (1) are not automatic; (2) do not admit a presentation by a finite complete rewriting system / do not have finite Gröbner–Shirshov bases.

- What can be said for other interesting examples of this kind?
  - The shifted Plactic monoid (Serrano (2009))
  - The hypoplactic monoid (Novelli (1998))
  - Given by permutation relations (F. Cedó, E. Jespers, J. Okniński (2010))
  - Plactic-growth-like monoids (Duchamp & Krob (1994))

- Explore the relationship with the Automaton algebras of:

  e.g. Our results $\Rightarrow$ Plactic and Chinese algebras are automaton algebras.
Appendix
Biautomaticity - formal definition

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 $u_i, v_i \in A$. 
Biautomaticity - formal definition

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 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.