Word automaticity of tree automatic ordinals is decidable

Martin Huschenbett

Institute of Theoretical Computer Science
Ilmenau University of Technology
Germany

Cambridge, UK
June 21st, 2012
Motivation and main result

Automatic structures (Büchi, Rabin, Khoussainov & Nerode, etc.)

Essence: Use **finite automata** on words or trees to **present structures**.

presentable by word automata  \(\leftrightarrow\)  presentable by tree automata

(e.g. \((\mathbb{N}, \times), \omega^\omega)\)
Motivation and main result

**Automatic structures (Büchi, Rabin, Khoussainov & Nerode, etc.)**

**Essence:** Use *finite automata* on words or trees to *present structures*.

<table>
<thead>
<tr>
<th>Presentable by word automata</th>
<th>Presentable by tree automata</th>
</tr>
</thead>
<tbody>
<tr>
<td>(e.g. (\mathbb{N}, \times), (\omega^\omega))</td>
<td></td>
</tr>
</tbody>
</table>

**Problem**

Given a presentation of some structure \(S\) by tree automata, is it decidable whether \(S\) is presentable by word automata?
Motivation and main result

Automatic structures (Büchi, Rabin, Khoussainov & Nerode, etc.)

Essence: Use finite automata on words or trees to present structures.

- Presentable by word automata
- Presentable by tree automata

(e.g. $(\mathbb{N}, \times), \omega\omega$)

Problem

Given a presentation of some structure $S$ by tree automata, is it decidable whether $S$ is presentable by word automata?

Theorem (H 2011)

Given a presentation of some ordinal $\alpha$ by tree automata, it is decidable whether $\alpha$ is presentable by word automata.
Recognising tree languages

Two $\Sigma$-trees for $\Sigma = \{a, b, c\}$:
Recognising tree languages

Two $\Sigma$-trees for $\Sigma = \{a, b, c\}$:

1. $\quad \quad a \quad b \quad c$
   $\quad b \quad c \quad a$
   $\quad c \quad a \quad c$

2. $\quad b \quad a \quad c$
   $\quad c \quad a \quad a$

Definition

A tree automaton $A$ is a finite state machine that
- processes trees bottom-up (from the leaves towards the root) and
- accepts a tree if the state reached at the root is accepting.

The tree language recognised by $A$ is the set of all accepted trees.
Recognising binary relations of trees

A pair \((t_1, t_2)\) of \(\Sigma\)-trees is encoded by a \((\Sigma \cup \{\Box\})^2\)-tree \(\langle t_1, t_2 \rangle\):

\[
\begin{array}{c}
\begin{tikzpicture}
  \node (a) at (0,0) {a};
  \node (b) at (-1,-1) {b};
  \node (c) at (-1,-2) {c};
  \node (d) at (1,-1) {b};
  \node (e) at (1,-2) {c};
  \node (f) at (2,0) {a};
  \node (g) at (2,-1) {c};
  \node (h) at (3,-2) {c};
  \node (i) at (4,-2) {c};
  \draw (a) -- (b);
  \draw (a) -- (c);
  \draw (b) -- (d);
  \draw (b) -- (e);
  \draw (a) -- (f);
  \draw (f) -- (g);
  \draw (f) -- (h);
  \draw (f) -- (i);
\end{tikzpicture}
\end{array},
\begin{array}{c}
\begin{tikzpicture}
  \node (a) at (0,0) {b};
  \node (b) at (1,0) {c};
  \node (c) at (2,0) {a};
  \node (d) at (0,-1) {c};
  \node (e) at (1,-1) {a};
  \node (f) at (2,-1) {c};
  \node (g) at (3,-2) {a};
  \node (h) at (4,-2) {b};
  \node (i) at (5,-2) {b};
  \node (j) at (6,-2) {c};
  \node (k) at (7,-2) {c};
  \draw (a) -- (b);
  \draw (a) -- (c);
  \draw (b) -- (d);
  \draw (b) -- (e);
  \draw (c) -- (f);
  \draw (f) -- (g);
  \draw (g) -- (h);
  \draw (g) -- (i);
  \draw (g) -- (j);
  \draw (g) -- (k);
\end{tikzpicture}
\end{array}
\end{array} = \begin{array}{c}
\begin{tikzpicture}
  \node (a) at (0,0) {ab};
  \node (b) at (1,0) {ba};
  \node (c) at (2,0) {cc};
  \node (d) at (0,-1) {b\Box};
  \node (e) at (1,-1) {c\Box};
  \node (f) at (2,-1) {ab};
  \node (g) at (3,-2) {ba};
  \node (h) at (4,-2) {\Box c};
  \node (i) at (5,-2) {\Box b};
  \node (j) at (6,-2) {c\Box};
  \node (k) at (7,-2) {c\Box};
  \draw (a) -- (b);
  \draw (a) -- (c);
  \draw (b) -- (d);
  \draw (b) -- (e);
  \draw (c) -- (f);
  \draw (f) -- (g);
  \draw (g) -- (h);
  \draw (g) -- (i);
  \draw (g) -- (j);
  \draw (g) -- (k);
\end{tikzpicture}
\end{array}
\end{array}
Recognising binary relations of trees

A pair \((t_1, t_2)\) of \(\Sigma\)-trees is encoded by a \((\Sigma \cup \{\Box\})^2\)-tree \(\langle t_1, t_2 \rangle\):

\[
\begin{align*}
\langle &b &a &c &b &a &c &b &a \rangle, \\
\langle &b &c &a &b &c &c &c &a \rangle
\end{align*}
\]

\[
\begin{align*}
\langle &a &b &c &a &b &c &c &b \rangle, \\
\langle &c &a &c &c &a &b &c &c \rangle
\end{align*}
\]

\[
\begin{align*}
\langle &b &\Box &c &\Box &a &\Box &c &\Box \rangle, \\
\langle &c &\Box &c &\Box &a &\Box &b &\Box \rangle
\end{align*}
\]

\[
\begin{align*}
\langle &\Box &a &\Box &c &\Box &b &\Box &c \rangle
\end{align*}
\]

Definition

A tree automaton \textbf{synchronously recognises} a binary relation \(R\) of trees if it recognises the tree language

\[
\{ \langle t_1, t_2 \rangle \mid (t_1, t_2) \in R \},
\]

i.e., all encodings of pairs in \(R\).
Presenting ordinals by automata

Definition

A pair \((A_L; A_\prec)\) of tree automata is a presentation of an ordinal \(\alpha\) if

- \(A_L\) recognises a tree language \(L\) and
- \(A_\prec\) synchronously recognises a binary relation \(\prec\) on \(L\)

such that \((L; \prec)\) is a well-ordering of type \(\alpha\).
Presenting ordinals by automata

**Definition**
A pair \((A_L; A_<)\) of tree automata is a presentation of an ordinal \(\alpha\) if
- \(A_L\) recognises a tree language \(L\) and
- \(A_<\) synchronously recognises a binary relation \(<\) on \(L\) such that \((L; <)\) is a well-ordering of type \(\alpha\).

**Definition**
A pair \((M_L; M_<)\) of word automata is a presentation of \(\alpha\) if...
Presenting ordinals by automata

**Definition**

A pair \((A_L; A_\prec)\) of tree automata is a **presentation** of an ordinal \(\alpha\) if

- \(A_L\) recognises a tree language \(L\) and
- \(A_\prec\) synchronously recognises a binary relation \(<\) on \(L\)

such that \((L; <)\) is a well-ordering of type \(\alpha\).

**Definition**

A pair \((M_L; M_\prec)\) of word automata is a **presentation** of \(\alpha\) if...

**Theorem (Delhommé 2001)**

Let \(\alpha\) be an ordinal.

1. \(\alpha\) is presentable by word automata if, and only if, \(\alpha < \omega^\omega\).
2. \(\alpha\) is presentable by tree automata if, and only if, \(\alpha < \omega^{\omega^\omega}\).
Main result

Theorem (H 2011)

Given a presentation \((A_L, A_<)\) of some ordinal \(\alpha\) by tree automata,

1. it is decidable whether \(\alpha\) is presentable by word automata and,
2. in case it is, one can compute such a presentation of \(\alpha\).
Slim tree languages

**Definition**

1. The **thickness** $\mathcal{O}(t)$ of a tree $t$ is the maximum number of nodes on any level in $t$. 

\[
\begin{align*}
\mathcal{O} & \left( \begin{array}{c}
  b & c & a \\
  b & c & a & b \\
  c & a & c & c
\end{array} \right) = 4 \\
\mathcal{O} & \left( \begin{array}{c}
  b & c \\
  a & b & c & a \\
  c & a & c
\end{array} \right) = 2
\end{align*}
\]
Slim tree languages

Definition

1. The thickness $\varnothing(t)$ of a tree $t$ is the maximum number of nodes on any level in $t$.
2. A tree language $L$ is **slim** if there exists a uniform upper bound on the thicknesses of all trees in $L$. Otherwise $L$ is **fat**.

$\varnothing\begin{pmatrix}
  \begin{array}{c}
    c \\
    a
  \end{array} & \begin{array}{c}
    \begin{array}{c}
      b \\
      c
    \end{array}
  \end{array} & \begin{array}{c}
    a
  \end{array} & \begin{array}{c}
    \begin{array}{c}
      c \\
      b
    \end{array}
  \end{array}
\end{pmatrix} = 4$

$\varnothing\begin{pmatrix}
  \begin{array}{c}
    a \\
    \begin{array}{c}
      b \\
      c
    \end{array}
  \end{array} & \begin{array}{c}
    \begin{array}{c}
      c \\
      a
    \end{array}
  \end{array} & \begin{array}{c}
    \begin{array}{c}
      a \\
      c
    \end{array}
  \end{array}
\end{pmatrix} = 2$
Main result

Theorem (H 2011)

Given a presentation \((A_L, A_<)\) of some ordinal \(\alpha\) by tree automata,

1. it is decidable whether \(\alpha\) is presentable by word automata and,
2. in case it is, one can compute such a presentation of \(\alpha\).

Proof sketch.

Does \(A_L\) recognise a slim tree language?
Main result

Theorem (H 2011)

Given a presentation \((A_L, A_<)\) of some ordinal \(\alpha\) by tree automata,

1. it is decidable whether \(\alpha\) is presentable by word automata and,
2. in case it is, one can compute such a presentation of \(\alpha\).

Proof sketch.

Decidable!

\[ \text{Does } A_L \text{ recognise a slim tree language?} \]
Main result

Theorem (H 2011)

Given a presentation \((A_L, A_\prec)\) of some ordinal \(\alpha\) by tree automata,
1. it is decidable whether \(\alpha\) is presentable by word automata and,
2. in case it is, one can compute such a presentation of \(\alpha\).

Proof sketch.

Decidable!

Does \(A_L\) recognise a slim tree language?

Yes

One can compute a presentation of \(\alpha\) by word automata.
Main result

Theorem (H 2011)

Given a presentation \((A_L, A_\prec)\) of some ordinal \(\alpha\) by tree automata,

1. it is decidable whether \(\alpha\) is presentable by word automata and,
2. in case it is, one can compute such a presentation of \(\alpha\).

Proof sketch.

Decidable!

Does \(A_L\) recognise a slim tree language?

Yes

One can compute a presentation of \(\alpha\) by word automata.

No

\(\alpha \geq \omega^\omega\), i.e., \(\alpha\) is not presentable by word automata.
Application of the main result

Theorem (Khoussainov, Nerode 1995/Blumensath 1999)

Let $S$ be structure presented by word/tree automata.
Every elementarily definable relation on $S$ is synchronously recognisable by a word/tree automaton.
Application of the main result

Theorem (Khoussainov, Nerode 1995/Blumensath 1999)

Let $S$ be structure presented by word/tree automata. Every elementarily definable relation on $S$ is synchronously recognisable by a word/tree automaton.

Theorem (H 2012)

Given a presentation by tree automata of some structure $S$ admitting an elementarily definable well-ordering of the domain,

1. it is decidable whether $S$ is presentable by word automata and,
2. in case it is, one can compute such a presentation of $S$. 
Summary and outlook

Theorem (H 2011)

Given a presentation of some ordinal $\alpha$ by tree automata, it is decidable whether $\alpha$ is presentable by word automata.

Open questions

▶ Given a presentation of some linear ordering $L$ by tree automata, is it decidable whether $L$ is presentable by word automata?

(New techniques are necessary: $(\mathbb{Q}; <)$ is presentable by word automata and by fat tree automata.)

▶ What about other classes of structures, e.g., graphs, groups, Boolean algebras, etc.?
Summary and outlook

**Theorem (H 2011)**

Given a presentation of some ordinal $\alpha$ by tree automata, it is decidable whether $\alpha$ is presentable by word automata.

**Theorem (H 2011)**

Given a presentation of a scattered linear ordering $\mathcal{L}$ by tree automata, it is decidable whether $\mathcal{L}$ is presentable by word automata.

Open questions

▶ Given a presentation of some linear ordering $\mathcal{L}$ by tree automata, is it decidable whether $\mathcal{L}$ is presentable by word automata?

(New techniques are necessary: $(\mathbb{Q}; <)$ is presentable by word automata and by fat tree automata.)

▶ What about other classes of structures, e.g., graphs, groups, Boolean algebras, etc.?
Summary and outlook

Theorem (H 2011)

Given a presentation of some ordinal $\alpha$ by tree automata, it is decidable whether $\alpha$ is presentable by word automata.

Theorem (H 2011)

Given a presentation of a scattered linear ordering $\mathcal{L}$ by tree automata, it is decidable whether $\mathcal{L}$ is presentable by word automata.

Open questions

- Given a presentation of some linear ordering $\mathcal{L}$ by tree automata, is it decidable whether $\mathcal{L}$ is presentable by word automata? (New techniques are necessary: $(\mathbb{Q}; <)$ is presentable by word automata and by fat tree automata.)
- What about other classes of structures, e.g., graphs, groups, Boolean algebras, etc.?