Restarting Tree Automata

Heiko Stamer and **Friedrich Otto**

Universität Kassel
Fachbereich Mathematik/Informatik
Heinrich-Plett-Straße 40, 34132 Kassel, Germany

stamer@theory.informatik.uni-kassel.de

76F7 3011 329D 27DB 8D7C 3F97 4F58 4EB8 FB2B E14F

16. Theorietag der GI-Fachgruppe 0.1.5
Technische Universität Wien, 28.-29. September 2006

UNIKASSEL VERSITÄT

UNI KASSEL
VERSITÄT

Agenda:

- Extend the computation model of restarting automata to a more complex data structures, i.e. trees resp. terms of a free algebra
  - Intention: String languages accepted by restarting automata should fit into the new tree automata model in a natural way (unary symbols)
  - Preserve the characteristic properties of the restarting automata (e.g. efficiently decidable membership problem, EPP, CPP)
- Establish new relationships and characterizations to well-known classes of tree automata and tree grammars
- Develop new applications for such restarting automata, e.g. verification of security protocols and tree transformations (XSLT)
- Top-down approach: Process the input tree beginning at the root and walk down to the leaves in parallel computation branches

Definition (Restarting Tree Automaton)

An RRWWT-automaton is a six-tuple $\mathcal{A} = (\mathcal{F}, \mathcal{G}, \mathcal{Q}, q_0, k, \Delta)$, where

- $\mathcal{F}$ is a ranked input alphabet, $\mathcal{G} \supseteq \mathcal{F}$ is a ranked working alphabet,
- $\mathcal{Q} = \mathcal{Q}_1 \cup \mathcal{Q}_2$ is a finite set of states (unary symb.) such that $\mathcal{Q}_1 \cap \mathcal{Q}_2 = \emptyset$,
- $q_0 \in \mathcal{Q}_1$ is the initial state and simultaneously the restart state,
- $k \geqslant 1$ is the height of the read/write window, and
- $\Delta = \Delta_1 \cup \Delta_2$ is a finite term rewriting system (TRS) on $\mathcal{G} \cup \mathcal{Q}$.

(Stateless) configuration: ground term from $\mathcal{T}(\mathcal{G} \cup \mathcal{Q})$ resp. $\mathcal{T}(\mathcal{G})$

$\to_\Delta \quad \to_{\Delta_1}$ move relation induced by TRS Δ resp. Δ_1 (appear within a cycle)

$\hookrightarrow_\mathcal{A}$ cycle relation, i.e., $u \hookrightarrow_\mathcal{A} v$ iff $q_0(u)(\to_\Delta^+ \setminus \to_{\Delta_1}^+)v$, for $u, v \in \mathcal{T}(\mathcal{G})$

$$L(\mathcal{A}) = \left\{ t_0 \in \mathcal{T}(\mathcal{F}) \;\middle|\; \begin{array}{l} \exists \ell \geqslant 0, \exists t_1, \ldots, t_\ell \in \mathcal{T}(\mathcal{G}) \text{ such that} \\ t_0 \hookrightarrow_\mathcal{A} t_1, \ldots, t_{\ell-1} \hookrightarrow_\mathcal{A} t_\ell, \text{ and } q_0(t_\ell) \to_{\Delta_1}^+ t_\ell \end{array} \right\}$$

Rule set Δ_1 contains only the following linear transition rules:

1. *k*-height bounded top-down transitions of the form

$$q(t) \to t[q_1(x_1), \ldots, q_m(x_m)],$$

where $m \geqslant 1$, $t \in \mathrm{Ctx}(\mathcal{G}, \mathcal{X}_m)$ is a nonempty m-context such that $1 \leqslant \mathrm{Hgt}(t) \leqslant k$, $x_1, \ldots, x_n \in \mathcal{X}_m$, and $q, q_1, \ldots, q_m \in \mathcal{Q}_1$.

2. *k*-height bounded final transitions of the form $q(t) \to t$, where $q \in \mathcal{Q}_1$ and $t \in \mathcal{T}(\mathcal{G})$ such that $0 \leqslant \mathrm{Hgt}(t) \leqslant k$.

Rule set Δ_2 (recall $\Delta = \Delta_1 \cup \Delta_2$) contains only the following linear rules:

1. k-height bounded top-down transitions of the form

$$q(t) \to t[q_1(x_1), \ldots, q_m(x_m)],$$

where $m \geqslant 1$, $t \in \mathrm{Ctx}(\mathcal{G}, \mathcal{X}_m)$ is a nonempty m-context such that $1 \leqslant \mathrm{Hgt}(t) \leqslant k$, $x_1, \ldots, x_n \in \mathcal{X}_m$, and $q, q_1, \ldots, q_m \in \mathcal{Q}_2$.

2. k-height bounded final transitions of the form $q(t) \to t$, where $q \in \mathcal{Q}_2$ and $t \in \mathcal{T}(\mathcal{G})$ such that $0 \leqslant \mathrm{Hgt}(t) \leqslant k$.

3. Size-reducing top-down rewrite transitions of the form

$$q(t) \to t'[q_1(x_1), \ldots, q_m(x_m)],$$

where $m \geqslant 1$, $t \in \mathcal{T}(\mathcal{G}, \mathcal{X}_m)$ is a linear term, $t' \in \mathrm{Ctx}(\mathcal{G}, \mathcal{X}_m)$ is a m-context, $x_1, \ldots, x_n \in \mathcal{X}_m$, $q \in \mathcal{Q}_1$, and $q_1, \ldots, q_m \in \mathcal{Q}_2$ such that $||t|| > ||t'||$ and $\mathrm{Hgt}(t) \leqslant k$.

4. Size-reducing final rewrite transitions of the form $q(t) \to t'$, where $q \in \mathcal{Q}_1$ and $t, t' \in \mathcal{T}(\mathcal{G})$ such that $||t|| > ||t'||$ and $\mathrm{Hgt}(t) \leqslant k$.

Restricted variants of a RRWWT-automaton:

RR-prefix: No restriction.

R-prefix: All top-down rewrite transitions have the special form

$$q(t) \to t'[x_1, \ldots, x_m],$$

i.e. all top-down/final transitions from Δ_2 are obsolete.

WW-suffix: No restriction.

W-suffix: The working alphabet $\mathcal{G}$ coincides with the input alphabet $\mathcal{F}$, i.e., there are no auxiliary symbols available.

λ-suffix: The right-hand side of every top-down/final rewrite transition is a "scattered subterm" of the corresponding left-hand side, that means, the right-hand side is homeomorphically embedded in the left-hand side.

Let $\mathcal{A} = (\mathcal{F}, \mathcal{G}, \mathcal{Q}, q_0, k, \Delta)$ be a RRWWT-automaton and $u, v \in \mathcal{T}(\mathcal{F})$:

- The TRS Δ is terminating and $\Delta^*(\{q_0(t) \mid t \in \mathcal{T}(\mathcal{G})\}) \in \mathscr{L}(\downarrow\text{NFT})$.
- The number of cycles performed by $\mathcal{A}$ during a computation on a given input tree $t \in \mathcal{T}(\mathcal{F})$ is upper-bounded by the size, i.e. $||t||$.
- Error Preserving Property (EPP):

  $$\text{If } u \hookrightarrow_{\mathcal{A}} v \text{ and } u \notin L(\mathcal{A}), \text{ then } v \notin L(\mathcal{A}).$$
- Furthermore, a weak pumping lemma holds.

Let $\mathcal{A}$ be a deterministic RRWWT-automaton, i.e. $\mathrm{CP}(\Delta) = \emptyset$:

- The TRS Δ is even convergent (terminating and confluent).
- Correctness Preserving Property (CPP):

  $$\text{If } u \hookrightarrow_{\mathcal{A}} v \text{ and } u \in L(\mathcal{A}), \text{ then } v \in L(\mathcal{A}).$$

Example

Let $\mathcal{F} = \mathcal{G} = \{f(\cdot,\cdot), g(\cdot), a\}$ be the ranked input resp. working alphabet, $\mathcal{Q} = \mathcal{Q}_1 \cup \mathcal{Q}_2$ the set of states with a partition $\mathcal{Q}_1 = \{q_0\}$ and $\mathcal{Q}_2 = \emptyset$, and $k = 2$ the height of the read/write window.
The TRS $\Delta = \Delta_1 \cup \Delta_2$ is given by the following transition rules:

$\Delta_1 : q_0(f(a,a)) \rightarrow f(a,a)$ $\Delta_2 : q_0(f(g(x_1), g(x_2))) \rightarrow f(x_1, x_2)$

The RT-automaton $\mathcal{A}_1 = (\mathcal{F}, \mathcal{G}, \mathcal{Q}, q_0, k, \Delta)$ recognizes the tree language $L_1 = \{f(g^n(a), g^n(a)) \mid n \geqslant 0\} \in \mathscr{L}(\text{CFTG}) \setminus \mathscr{L}(\downarrow\text{NFT})$.

There exists a det-RT-automaton which recognizes $\{f(a,b), f(b,a)\}$.

Corollary

We have $\mathscr{L}(\downarrow NFT) \subsetneq \mathscr{L}(RT)$ and $\mathscr{L}(\downarrow DFT) \subsetneq \mathscr{L}(det\text{-}RT)$.

Example (RT-automaton, language of completely balanced binary trees)

Let $\mathcal{F} = \mathcal{G} = \{f(\cdot,\cdot), a\}$ be the input/working alphabet, $\mathcal{Q} = \mathcal{Q}_1 \cup \mathcal{Q}_2$ the set of states with $\mathcal{Q}_1 = \{q_0, q_1\}$ and $\mathcal{Q}_2 = \emptyset$, $k = 1$ the height of the read/write window, and the TRS $\Delta = \Delta_1 \cup \Delta_2$ as follows:

Δ_1 : $q_0(f(a,a)) \to f(a,a)$ $\quad$ $q_0(a) \to a$ $\quad$ (final/top-down transitions)

$q_0(f(x_1,x_2)) \to f(q_1(x_1), q_1(x_2))$ $\quad$ $q_1(f(x_1,x_2)) \to f(q_1(x_1), q_1(x_2))$

Δ_2 : $q_1(f(a,a)) \to a$ $\quad$ (size-reducing final rewrite transition)

Example ($L_4 = \{f(g^n(h^n(a)), g^n(h^n(a))) \mid n \geqslant 1\} \notin \mathscr{L}(\text{CFTG})$)

Let $\mathcal{F} = \mathcal{G} = \{f(\cdot,\cdot), g(\cdot), h(\cdot), a\}$ be the ranked input/working alphabet, $\mathcal{Q} = \mathcal{Q}_1 \cup \mathcal{Q}_2$ the set of states with $\mathcal{Q}_1 = \{q_0, q_1\}$ and $\mathcal{Q}_2 = \emptyset$, $k = 3$ the height of the read/write window, and $\Delta = \Delta_1 \cup \Delta_2$ as follows:

Δ_1 : $q_0(f(g(h(a)), g(h(a)))) \to f(g(h(a)), g(h(a)))$ (final/top-down trans.)

$q_0(f(g(x_1), g(x_2))) \to f(g(q_1(x_1)), g(q_1(x_2)))$ $\quad q_1(g(x_1)) \to g(q_1(x_1))$

Δ_2 : $q_1(g(h(h(x_1)))) \to h(x_1)$ (size-reducing top-down rewrite transition)

Corollary

Even $\mathscr{L}(RT)$ contains tree languages that are not context-free.

Proposition

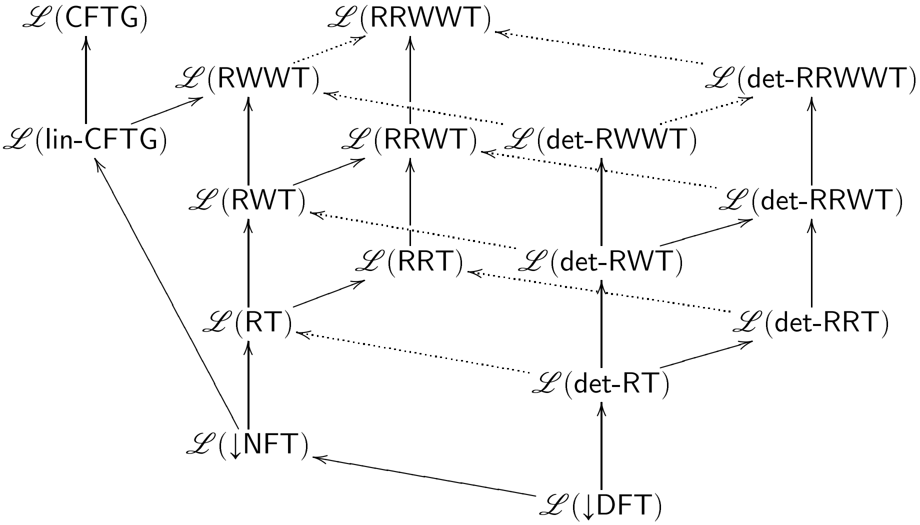
For every linear context-free tree grammar G there exists an equivalent linear nondeleting growing tree grammar G' such that $L(G) = L(G')$.

Proposition

For every linear nondeleting growing context-free tree grammar G there exists a RWWT-automaton $\mathcal{A}$ such that $L(G) = L(\mathcal{A})$.

Corollary

The inclusion $\mathscr{L}(\text{lin-CFTG}) \subseteq \mathscr{L}(\text{RWWT})$ is proper.



$\mathscr{L}$(CFTG)
$\mathscr{L}$(RRWWT)
$\mathscr{L}$(RWWT)
$\mathscr{L}$(det-RRWWT)
$\mathscr{L}$(lin-CFTG)
$\mathscr{L}$(RRWT)
$\mathscr{L}$(det-RWWT)
$\mathscr{L}$(RWT)
$\mathscr{L}$(det-RRWT)
$\mathscr{L}$(RRT)
$\mathscr{L}$(det-RWT)
$\mathscr{L}$(RT)
$\mathscr{L}$(det-RRT)
$\mathscr{L}$(det-RT)
$\mathscr{L}$(↓NFT)
$\mathscr{L}$(↓DFT)

Applying the proof technique of Niemann und Otto [NO03] we obtain:

Proposition

A tree language L is recognized by an RRWWT-/RWWT-automaton if and only if there exists an RRWT-/RWT-automaton $\mathcal{A}$ and a $\downarrow$NFT-automaton $\mathcal{B}$ such that $L = L(\mathcal{A}) \cap L(\mathcal{B})$.

Unfortunately, this result is not as general as for restarting automata.

Corollary

$\mathscr{L}(\text{RWWT})$ and $\mathscr{L}(\text{RRWWT})$ are closed under intersection with regular tree languages.

Proposition

$\mathscr{L}(\text{RWWT})$ and $\mathscr{L}(\text{RRWWT})$ are closed under union.

Open Questions:

- Does $\mathscr{L}(\text{CFTG}) \subsetneq \mathscr{L}(\text{RWWT})$ resp. $\subsetneq \mathscr{L}(\text{RRWWT})$ holds?
- Closure under linear tree homomorphisms/transductions
- Restricted variants with decidable emptiness problem
- Yield and path languages of restarting tree automata
- Restarting tree automata on unranked alphabets (XSLT)

Thank you! Questions?

Hubert Comon, Max Dauchet, Rémi Gilleron, Florent Jacquemard, Denis Lugiez, Sophie Tison, Marc Tommasi.
Tree Automata Techniques and Applications.
http://www.grappa.univ-lille3.fr/tata/

Friedrich Otto.
Restarting Automata and their Relations to the Chomsky Hierarchy.
Proceedings DLT 2003, LNCS 2710, pp. 55–74, 2003.

Friedrich Otto.
Restarting Automata.
Notes for a Course at the 3rd International PhD School in Formal Languages and Applications, 2004.

Gundula Niemann, Friedrich Otto.
Further results on restarting automata.
Proceedings of Words, Languages and Combinatorics III, pp. 352–369, 2003.

Thomasz Jurdziński, Krzysztof Loryś, Gundula Niemann, Friedrich Otto.
Some results on RWW- and RRWW-automata and their relationship to the class of growing context-sensitive languages.
Journal of Automata, Languages and Combinatorics, 9:407–437, 2004.