Subtree and Tree Pattern Pushdown Automata for Trees In Prefix Notation

Jan Janoušek and Bořivoj Melichar
{janousej,melichar}@fel.cvut.cz

Department of Computer Science and Engineering
Faculty of Electrical Engineering
Czech Technical University
Czech Republic
MOTIVATION

STRINGOLOGY
String suffix and factor automata.

PROPERTIES:

1. **Accept all occurrences of an input suffix and an input factor, respectively, in a text of size $n$.**

2. **Search phase for all occurrences of an input suffix or an input factor of size $m$ in time $O(m)$, and not depending on $n$.**

3. **Although the number of factors in the text can be quadratic in $n$, the total size of the deterministic factor automaton is linear in $n$.**
MOTIVATION

ARBOLOGY

Subtree and tree pattern pushdown automata — ANALOGOUS TO STRING SUFFIX AND FACTOR AUTOMATA.

PROPERTIES:

1. Accept all occurrences of an input subtree and of subtrees matching an input tree pattern, respectively, in a tree of size $n$.

2. Search phase for all occurrences of an input subtree or an input tree pattern of size $m$ in time $O(m)$, and not depending on $n$.

3. Although the number of tree patterns matching the tree can be exponential in $n$, the total size of the deterministic tree pattern pushdown automaton is linear in $n$. 
**SUBTREE PDA**

**EXAMPLE 1**

- **Ranked alphabet**
  \[ A = \{ a2, a1, a0 \} \]

- **Tree** \( t_1 \)
  Prefix notation is
  \[ \text{pref}(t_1) = a2 \ a2 \ a0 \ a1 \ a0 \ a1 \ a0 \]

- **Subtrees of** \( t_1 \) **in**
  Prefix notation are:
  \[
  \begin{align*}
  1 & : a2 \ a2 \ a0 \ a1 \ a0 \ a1 \ a0 \\
  2 & : a2 \ a0 \ a1 \ a0 \\
  3 & : a1 \ a0 \\
  4 & : a0
  \end{align*}
  \]
ALL SUBTREES OF TREE $t_1$ AND THEIR PREFIX NOTATION

```
   a2
   / \
  a2 a1
 /     / \
a0    a0 a0

   a2
   / \
  a2 a0
 /     / \
a0    a1 a0

   a2
 / \
 a0 a1
 /     / \
 a0 a0 a0

   a2
 / \
 a2 a0
 /     / \
 a0 a1 a0

   a1
 / \
 a1 a0
 /     / \
 a0 a0 a0

   a1
   / \
   a1
    /     / \
   a0 a0 a0
```

$a2$ $a2$ $a0$ $a1$ $a0$ $a1$ $a0$

$a2$ $a0$ $a1$ $a0$

$a1$ $a0$ $a0$

$a1$ $a0$

$a0$
**Theorem 1**

**Given a tree** $t$ **and its prefix notation** $\text{pref}(t)$, **all subtrees of** $t$ **in prefix notation are substrings of** $\text{pref}(t)$.
Example 1, contd.

Transition diagram of deterministic PDA $M_p(t_1)$ accepting $\text{pref}(t_1) = a2 \ a2 \ a0 \ a1 \ a0 \ a1 \ a0$ by empty pushdown store

Initial contents of pushdown store is $S$. 
TRACE OF DETERMINISTIC PDA $M_p(t_1)$ FOR INPUT STRING $\text{pref}(t_1) = a2\ a2\ a0\ a1\ a0\ a1\ a0$

<table>
<thead>
<tr>
<th>State</th>
<th>Pushdown Store</th>
<th>Input</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>S</td>
<td>$a2\ a2\ a0\ a1\ a0\ a1\ a0$</td>
</tr>
<tr>
<td>1</td>
<td>S S</td>
<td>$a2\ a0\ a1\ a0\ a1\ a0$</td>
</tr>
<tr>
<td>2</td>
<td>S S S</td>
<td>$a0\ a1\ a0\ a1\ a0$</td>
</tr>
<tr>
<td>3</td>
<td>S S S</td>
<td>$a1\ a0\ a1\ a0$</td>
</tr>
<tr>
<td>4</td>
<td>S S S</td>
<td>$a0\ a1\ a0$</td>
</tr>
<tr>
<td>5</td>
<td>S</td>
<td>$a1\ a0$</td>
</tr>
<tr>
<td>6</td>
<td>S</td>
<td>$a0$</td>
</tr>
<tr>
<td>7</td>
<td>$\varepsilon$</td>
<td>$\varepsilon$</td>
</tr>
</tbody>
</table>

ACCEPT

ACCEPT BY EMPTY PUSHDOWN STORE.
Nondeterministic subtree PDA $M_{nps}(t_1)$ for tree $t_1$ in prefix notation

$\text{pref}(t_1) = a2 a2 a0 a1 a0 a1 a0$
TRANSFORMATION TO DETERMINISTIC PDA

INPUT-DRIVEN PDA – pushdown store operations are determined by the input symbol.

Any nondeterministic input–driven PDA can be determinised similarly as in the case of finite automata – the states of the deterministic PDA correspond to subsets of states of the nondeterministic PDA (d–subsets).

Moreover, nondeterministic acyclic input–driven PDA – the contents of the pushdown store can be precomputed, and only transitions and states with possible pushdown operations are selected.
Deterministic subtree PDA $M_{dps}(t_1)$ for tree in prefix notation $\text{pref}(t_1) = a_2 \ a_2 \ a_0 \ a_1 \ a_0 \ a_1 \ a_0$
Trace of deterministic subtree PDA $M_{dps}(t_1)$ for an input subtree $st$ in prefix notation

$pref(st) = a_1 a_0$

<table>
<thead>
<tr>
<th>STATE</th>
<th>PDS</th>
<th>INPUT</th>
<th>INPUT SUBTREE</th>
</tr>
</thead>
<tbody>
<tr>
<td>{0}</td>
<td>S</td>
<td>$a_1$</td>
<td>a1</td>
</tr>
<tr>
<td>{4, 6}</td>
<td>S</td>
<td>$a_0$</td>
<td>a0</td>
</tr>
<tr>
<td>{5, 7}</td>
<td>$\varepsilon$</td>
<td>$\varepsilon$</td>
<td></td>
</tr>
<tr>
<td>ACCEPT</td>
<td></td>
<td></td>
<td>a0</td>
</tr>
</tbody>
</table>
**TREE PATTERN PDA**

**Deterministic treetop PDA** $M_{pt}(t_1)$ **for tree** $t_1$ **in prefix notation** $\text{pref}(t_1) = a_2 \ a_2 \ a_0 \ a_1 \ a_0 \ a_1 \ a_0$
Nondeterministic tree pattern PDA $M_{npg}(t_1)$ for $\text{pref}(t_1) = a2\ a2\ a0\ a1\ a0\ a1\ a0$
Deterministic tree pattern PDA $M_{dpq}(t_1)$ for tree $t_1$ in prefix notation

$\text{pref}(t_1) = a_2 \; a_2 \; a_0 \; a_1 \; a_0 \; a_1 \; a_0$
Trace of deterministic PDA $M_{dpg}$ for prefix notation of tree pattern $a_2 \text{ } S \text{ } a_1 \text{ } S$

<table>
<thead>
<tr>
<th>STATE</th>
<th>PDS</th>
<th>INPUT</th>
</tr>
</thead>
<tbody>
<tr>
<td>{0}</td>
<td>S</td>
<td>$a_2 \text{ } S \text{ } a_1 \text{ } S$</td>
</tr>
<tr>
<td>{1, 2}</td>
<td>SS</td>
<td>$S \text{ } a_1 \text{ } S$</td>
</tr>
<tr>
<td>{3, 5}</td>
<td>S</td>
<td>$a_1 \text{ } S$</td>
</tr>
<tr>
<td>{4, 6}</td>
<td>S</td>
<td>S</td>
</tr>
<tr>
<td>{5, 7}</td>
<td>$\varepsilon$</td>
<td>$\varepsilon$</td>
</tr>
<tr>
<td>ACCEPT</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Input tree template:

```
    a2
   /   |
S    a1
     /|
     S
```
Theorem 3

Given a tree $t$ with $n$ nodes and its prefix notation $\text{pref}(t)$, the numbers of states and transitions of the deterministic tree pattern PDA $M_{dpg}(t)$ are $\mathcal{O}(n)$.

(Although the number of distinct tree templates matching the tree is less or equal $2^{n-1}$)
**Example 2**

TREE $t_2$, $\text{pref}(t_2) = am a0^m$

Deterministic tree pattern PDA for $\text{pref}(t_2)$:
**Example 3**

TREE $t_3$, $\text{pref}(t_3) = a_1^m a_0$

**Deterministic tree pattern PDA for $\text{pref}(t_3)$:**

```
S | S \rightarrow \varepsilon
S | S \rightarrow \varepsilon
S | S \rightarrow \varepsilon
S | S \rightarrow \varepsilon
```

```
S \rightarrow S
S \rightarrow S
S \rightarrow S
S \rightarrow S
```

```
as_1 \rightarrow S
as_1 \rightarrow S
as_1 \rightarrow S
as_1 \rightarrow S
```

```
\{0\} \rightarrow \{1, 2, 3, \ldots, m\} \rightarrow \{2, 3, \ldots, m\} \rightarrow \{3, \ldots, m\} \rightarrow \ldots \rightarrow \{m\} \rightarrow \{m + 1\}
```

```
as_1 \rightarrow S
a_0 | S \rightarrow \varepsilon
a_1 | S \rightarrow S
```

```
as_0 \rightarrow \varepsilon
```

```
\varepsilon
\varepsilon
\varepsilon
\varepsilon
```
WEB PAGES
http://www.arbology.org
http://www.arbology.com
COMING SOON ...
Tree Languages, Tree Automata and Deterministic Pushdown Automata

Regular tree languages are accepted by finite tree automata.

Deterministic pushdown automata accept a proper superclass of the regular tree languages in prefix or postfix notation.

This is proved in:

This paper contains also algorithm of transformation of any finite tree automaton to an equivalent deterministic pushdown automaton.