Enumeration on Trees with Tractable Combined Complexity and Efficient Updates

Antoine Amarilli, Pierre Bourhis, Stefan Mengel, Matthias Niewerth

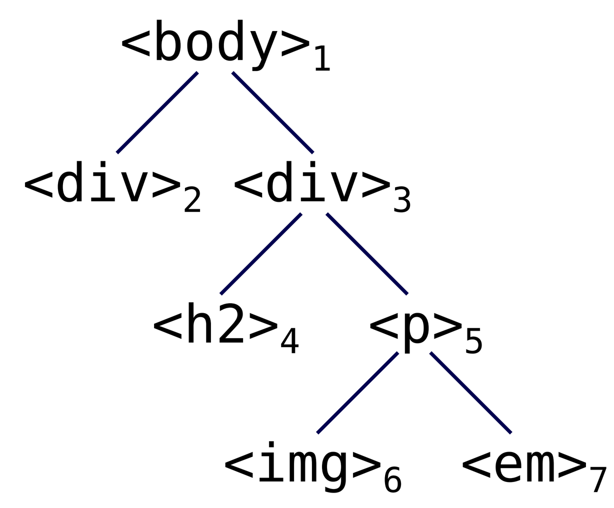
Problem Description

Compute all answers of an **MSO query** given by a tree automaton on an input **tree** and maintain them efficiently under **updates**

- **MSO** = **first-order logic + quantification over sets**
  Example: *find all pairs of an* ***h2 section title*** *and of an* ***image*** *located in the same section*
- **Tree** with nodes labeled in a **fixed finite alphabet**
- **Updates** are **local**: relabelings, insertion & removal of leaves
- **Complexity** is polynomial in the automaton and linear in the tree

Example

Example tree:



Example answer set: {⟨4, 6⟩}

Example updates:

- **Add** new node <img>8 below node 7
  New answer set: {⟨4, 6⟩, ⟨4, 8⟩}
- **Delete** node 4
  New answer set: ∅

Main Result and Existing Work Comparison

Theorem (Bagan in 2006; Kazana and Segoufin in 2013):
Enumeration of **answers** to an MSO query on a tree is possible with **linear** preprocessing and **constant** delay (**linear delay** in the size of answers for free second order variables)

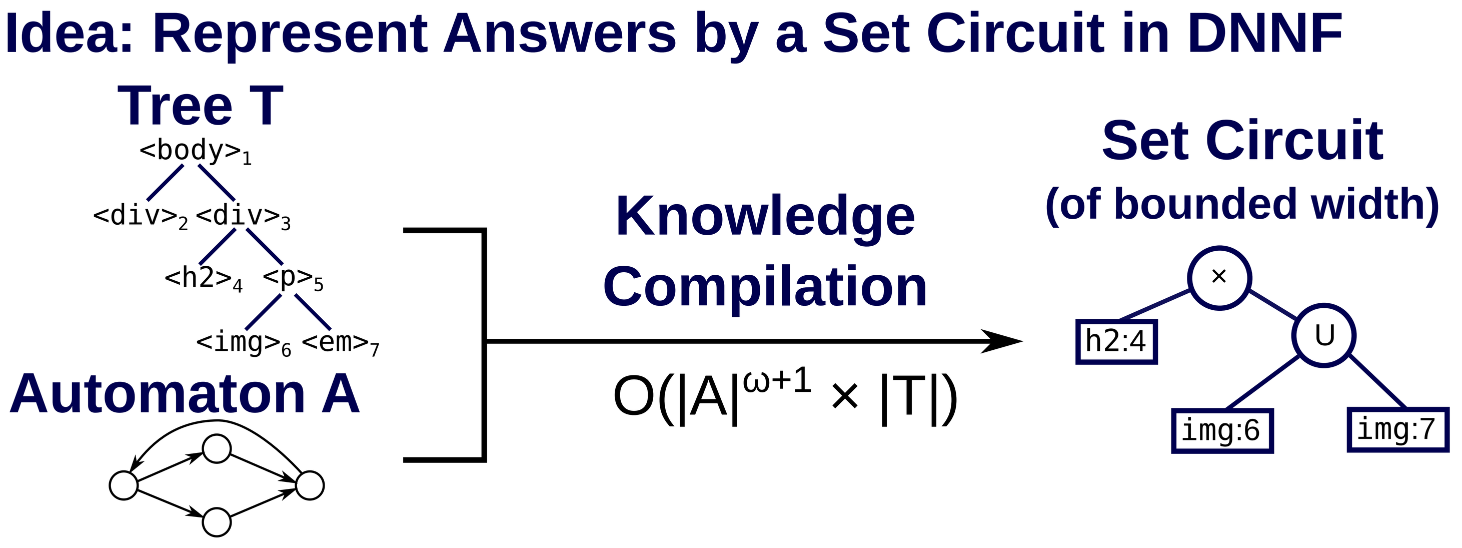
Our main results:

- We can handle local **updates** in log time:
  relabeling of nodes, insertion & deletion of leaves
- Our algorithm is **polynomial** in the automaton

Work	Delay	Updates
Bagan'06 and Kazana&Segoufin '13	$O(1)$	$O(\|T\|)$: re-index the tree
Losemann&Martens'14	$O(\log^2\|T\|)$	$O(\log^2\|T\|)$
Niewerth&Segoufin'18	$O(1)$	$O(\log \|T\|)$, only on strings
Amarilli et al.'18	$O(1)$	$O(\log \|T\|)$, only relabelings
Our work	$O(1)$	$O(\log \|T\|)$

Proof Approach: Knowledge Compilation

Idea: Represent Answers by a Set Circuit in DNNF

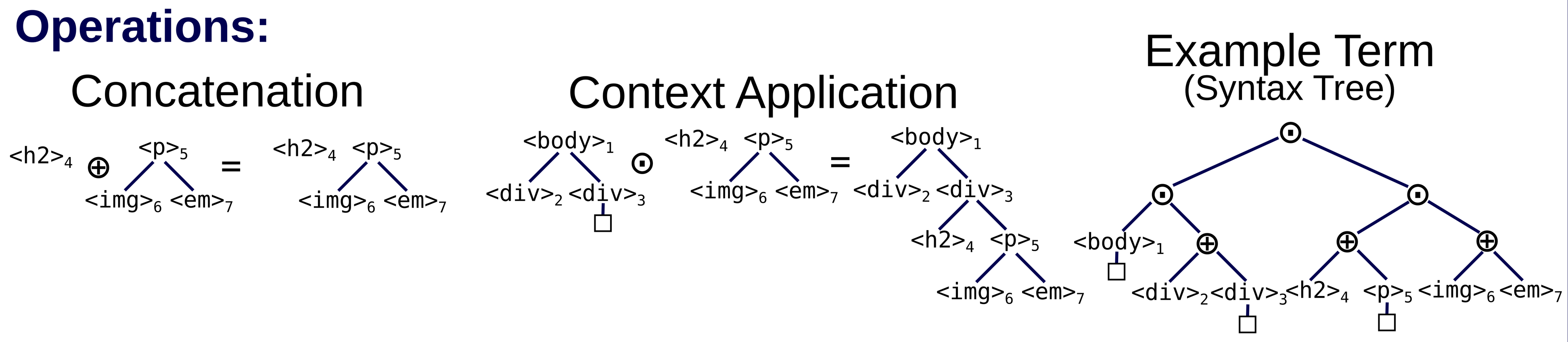


Gate	Symbol	Semantics	
Variable	img:6	{ {img:6} }	
Union	∪	⟦G₁⟧ ∪ ⟦G₂⟧	with input Gates G₁ and G₂
Product	×	{ A₁ ∪ A₂ | Aᵢ ∈ ⟦Gᵢ⟧ }	

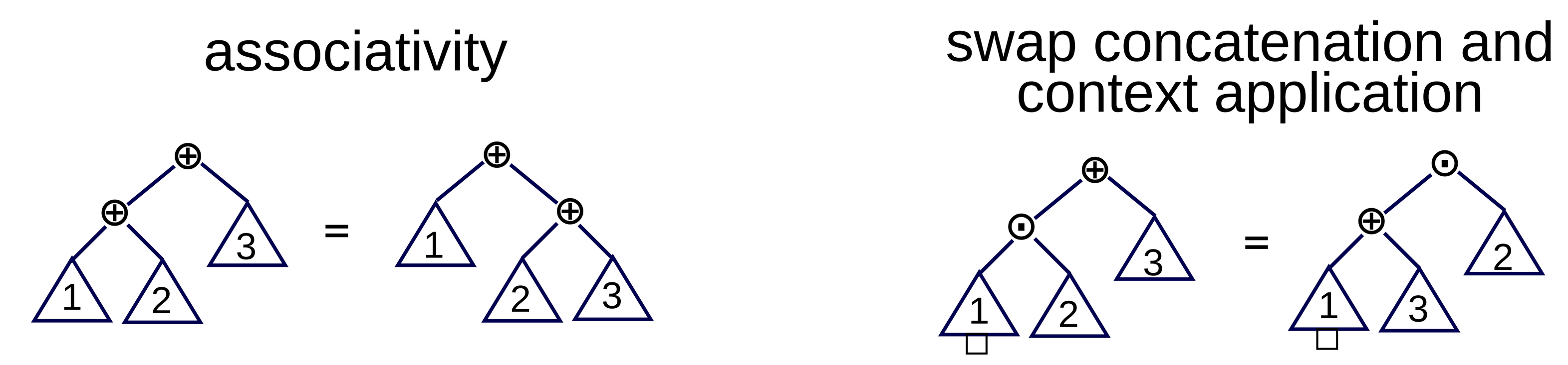
Proof Tool: Forest Algebras

Idea: Represent trees by syntax trees in the free forest algebra

Operations:



Rewritings (similar to rotations in AVL trees)



Algorithm