**The Emptiness Problem for Alternating Finite Memory Automata**

Daniel Genkin, Michael Kaminski, **Liat Peterfreund**

---

### Introduction

One of the most important developments in worldwide web is the emergence of the Extended Markup Language (XML). XML is the standard data exchange format currently used in the web. Rather than using the traditional tabular form, XML represents data as a labeled tree. Almost any data format can easily be translated to XML in a transparent manner. In recent years, tree automata became one of the major tools used in the XML research. However, most abstractions in this direction ignore an important aspect of XML, namely, the presence of infinitely many data values attached to the nodes of trees.

**XML Basics**

An XML document can be viewed as a tree in a natural way.

- **Factory**
  - Name:“Monstropolis”
  - Product (id=901)
  - Product (id=904)
  - Product (id=9011)
- **Advertisement**
  - Name
  - No.
  - Close door
  - Scream animation

The structure of an XML document is described in a suitable DTD file:

```xml
<!DOCTYPE factory [ 
  <!ELEMENT factory (production, advertisement) >
  <!ELEMENT production (product)* >
  <!ELEMENT advertisement (product)! >
  <!ELEMENT product (No.,name) | ε >
  <!ELEMENT No. (#PCDATA) >
  <!ELEMENT name (#PCDATA) >
  <!ATTLIST factory name #PCDATA >
  <!ATTLIST factory product id #PCDATA >
  ]>
```

---

### Model Description

Let $L$ be the following language:

- $\#\sigma$ on every path is even
- $\#\#\sigma$ on every path is odd

Let us present an automaton $A$ such that $L(A) = L$:

![Diagram of automaton $A$]

The automaton’s run can be described as follows:

1. For each word $w \in \Sigma^*$ do:
   1. Run $A$ on input $w$.
   2. If $A(w) = \text{accept}$ then return ‘not empty’
2. Return ‘empty’

It might not terminate for the following problems with step 1.

- **Problem 1**: There are infinitely many $w \in \Sigma^*$ (even if $\Sigma$ is finite).
  - **Solution 1**: Find an upper bound $N$ such that if $L(A) \neq \emptyset$ then there exists $w \in L(A)$ and $|w| \leq N$.
- **Problem 2**: The alphabet is infinite.
  - **Solution 2**: Find a subset $\Sigma' \subseteq \Sigma$ such that if $L(A) \neq \emptyset$ then it contains a word over $\Sigma'$.

---

### The Question

Is there an algorithm that receives as an input an automaton and returns whether or not its language is empty?

**Main result**

The emptiness problem for one register alternating finite memory automata is decidable.

**A Naive Attempt**

Consider the following computation procedure:

- Is-$\text{Empty-Language}(A)$:
  1. For each word $w \in \Sigma^*$ do:
     1. Run $A$ on input $w$.
     2. If $A(w) = \text{accept}$ then return ‘not empty’
  2. Return ‘empty’

**Problem**: There are infinitely many $w \in \Sigma^*$ (even if $\Sigma$ is finite).

**Solution 1**: Find an upper bound $N$ such that if $L(A) \neq \emptyset$ then there exists $w \in L(A)$ and $|w| \leq N$.

**Problem 2**: The alphabet is infinite.

**Solution 2**: Find a subset $\Sigma' \subseteq \Sigma$ such that if $L(A) \neq \emptyset$ then it contains a word over $\Sigma'$.

---

### Restriction to finite alphabets

Indistinguishability property:

- The symbols occurring in the input are of no real significance.
- Only the initial and repetition pattern matter:

```
- A configuration can be described up to an automorphism $\alpha: \Sigma \rightarrow \Sigma$.
- Define a node to be a series of sequential configurations and a successor relation on the set of nodes. This impose a tree structure.
- If at any depth there exists a node that is irreducible then by an extension of König's Infinity Lemma there exists an infinite path of irreducible nodes.
- This contradicts Karp-Miller’s Lemma. Thus, there exists a reducible node at depth $N$.
```

**Upper bound on the minimal word length**

- Define a node to be a series of sequential configurations and a successor relation on the set of nodes. This impose a tree structure.
- If at any depth there exists a node that is irreducible then by an extension of König's Infinity Lemma there exists an infinite path of irreducible nodes.
- This contradicts Karp-Miller’s Lemma. Thus, there exists a reducible node at depth $N$.

**Resulting-Miller’s Lemma**

- There exists a reducible node at depth $N$.

**Is-$\text{Empty-Language}(A)$**:

1. For each word $w \in \Sigma^*$ do:
   1. Run $A$ on input $w$.
   2. If $A(w) = \text{accept}$ then return ‘not empty’
2. Return ‘empty’.

---

**Conclusion**

The emptiness problem for one register alternating finite memory automata automata is decidable.