Theoretical automata theory: $\oplus$-NFAs

- Succinctness
- Minimization and compression

Practical work: assistive technologies

- English text to South African Sign Language // sign recognition
- Assistive software for autism – games for speech therapy, cognitive robotics
Theoretical automata theory: $\oplus$-NFAs
- Succinctness
  - Minimization and compression

Practical work: assistive technologies
- English text to South African Sign Language // sign recognition
- Assistive software for autism – games for speech therapy, cognitive robotics
● Theoretical automata theory: ⊕-NFAs
  ● Succinctness
  ● Minimization and compression

● Practical work: assistive technologies
  ● English text to South African Sign Language // sign recognition
  ● Assistive software for autism – games for speech therapy, cognitive robotics
Shameless advertising

- Theoretical automata theory: $\oplus$-NFAs
  - Succinctness
  - Minimization and compression
- Practical work: assistive technologies
  - English text to South African Sign Language // sign recognition
  - Assistive software for autism – games for speech therapy, cognitive robotics
Theoretical automata theory: $\oplus$-NFAs
- Succinctness
- Minimization and compression

Practical work: assistive technologies
- English text to South African Sign Language // sign recognition
- Assistive software for autism – games for speech therapy, cognitive robotics
Theoretical automata theory: \(\oplus\)-NFAs
- Succinctness
- Minimization and compression

Practical work: assistive technologies
- English text to South African Sign Language // sign recognition
- Assistive software for autism – games for speech therapy, cognitive robotics
Overview

- NFAs and $\oplus$-NFAs
  - Ambiguity: definition, importance
  - Ambiguity in $\oplus$-NFAs
Overview

- NFAs and $\oplus$-NFAs
- Ambiguity: definition, importance
- Ambiguity in $\oplus$-NFAs
Overview

- NFAs and $\oplus$-NFAs
- Ambiguity: definition, importance
- Ambiguity in $\oplus$-NFAs
NFA reminder

An NFA is a finite state machine that, on reading its input string, can nondeterministically decide to which of its possible next states it wants to move. *Accepts language.*
An NFA is a finite state machine that, on reading its input string, can nondeterministically decide to which of its possible next states it wants to move. *Accepts language.*
An NFA is a finite state machine that, on reading its input string, can nondeterministically decide to which of its possible next states it wants to move. *Accepts language.*
An NFA is a finite state machine that, on reading its input string, can nondeterministically decide to which of its possible next states it wants to move. *Accepts language.*
An NFA is a finite state machine that, on reading its input string, can nondeterministically decide to which of its possible next states it wants to move. **Accepts language.**
What is a ⊕-NFA?
Parity machine – in execution tree, takes XOR of possible choices.

Why ⊕-NFAs?
- Sequencing versus ringlike repetition
- Hardware implementation as LFSR
- Regular languages distribution
What is a $\oplus$-NFA?

Parity machine – in execution tree, takes XOR of possible choices.

Why $\oplus$-NFAs?

- Sequencing versus ringlike repetition
- Hardware implementation as LFSR
- Regular languages distribution
What is a $\oplus$-NFA?
Parity machine – in execution tree, takes XOR of possible choices.

Why $\oplus$-NFAs?
- Sequencing versus ringlike repetition
- Hardware implementation as LFSR
- Regular languages distribution
What is a $\oplus$-NFA?

Parity machine – in execution tree, takes XOR of possible choices.

Why $\oplus$-NFAs?

- Sequencing versus ringlike repetition
- Hardware implementation as LFSR
- Regular languages distribution
What is ambiguity?

Given an NFA $M$, we define the ambiguity of a string $w$ to be the number of different accepting paths for $w$ in $M$. Function $amb_M(n)$ is max of ambiguities of strings of length $n$ or less.

Types of ambiguity

- unambiguous: ambiguity of any string is zero or one
- finitely ambiguous: $amb_M(n)$ bounded by constant function
- polynomially ambiguous: $amb_M(n)$ bounded by polynomial function
- exponentially ambiguous: $amb_M(n)$ bounded by exponential function
What is ambiguity?

Given an NFA $M$, we define the ambiguity of a string $w$ to be the number of different accepting paths for $w$ in $M$. Function $amb_M(n)$ is max of ambiguities of strings of length $n$ or less.

Types of ambiguity

- unambiguous: ambiguity of any string is zero or one
- finitely ambiguous: $amb_M(n)$ bounded by constant function
- polynomially ambiguous: $amb_M(n)$ bounded by polynomial function
- exponentially ambiguous: $amb_M(n)$ bounded by exponential function
What is ambiguity?
Given an NFA $M$, we define the ambiguity of a string $w$ to be the number of different accepting paths for $w$ in $M$. Function $amb_M(n)$ is max of ambiguities of strings of length $n$ or less.

Types of ambiguity
- unambiguous: ambiguity of any string is zero or one
- finitely ambiguous: $amb_M(n)$ bounded by constant function
- polynomially ambiguous: $amb_M(n)$ bounded by polynomial function
- exponentially ambiguous: $amb_M(n)$ bounded by exponential function
What is ambiguity?

Given an NFA $M$, we define the ambiguity of a string $w$ to be the number of different accepting paths for $w$ in $M$. Function $amb_M(n)$ is max of ambiguities of strings of length $n$ or less.

Types of ambiguity

- unambiguous: ambiguity of any string is zero or one
- finitely ambiguous: $amb_M(n)$ bounded by constant function
- polynomially ambiguous: $amb_M(n)$ bounded by polynomial function
- exponentially ambiguous: $amb_M(n)$ bounded by exponential function
Ambiguity

What is ambiguity?

Given an NFA $M$, we define the ambiguity of a string $w$ to be the number of different accepting paths for $w$ in $M$. Function $amb_M(n)$ is max of ambiguities of strings of length $n$ or less.

Types of ambiguity

- unambiguous: ambiguity of any string is zero or one
- finitely ambiguous: $amb_M(n)$ bounded by constant function
- polynomially ambiguous: $amb_M(n)$ bounded by polynomial function
- exponentially ambiguous: $amb_M(n)$ bounded by exponential function
<table>
<thead>
<tr>
<th>Why is ambiguity important?</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Nondeterminism is a <strong>resource</strong> of a finite automaton.</td>
</tr>
<tr>
<td>- Does the size of an NFA vary as the amount of ambiguity increases?</td>
</tr>
<tr>
<td>- (related to number of nondeterministic moves vs size)</td>
</tr>
</tbody>
</table>
**Why is ambiguity important?**

- Nondeterminism is a **resource** of a finite automaton.
- Does the size of an NFA vary as the amount of ambiguity increases?
  - (related to number of nondeterministic moves vs size)
Why is ambiguity important?

- Nondeterminism is a **resource** of a finite automaton.
- Does the size of an NFA vary as the amount of ambiguity increases?
- *(related to number of nondeterministic moves vs size)*
Examples of ambiguity in $\oplus$-NFAs

Unambiguous $\oplus$-NFA

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$a$</td>
<td></td>
</tr>
<tr>
<td>$q_0$</td>
<td>${q_0, q_1}$</td>
</tr>
<tr>
<td>$q_1$</td>
<td>${q_2}$</td>
</tr>
<tr>
<td>$q_2$</td>
<td>${q_0}$</td>
</tr>
</tbody>
</table>
Examples of ambiguity in $\oplus$-NFAs

Unambiguous $\oplus$-NFA

- $q_0$ to $q_0$ with label $a$
- $q_0$ to $q_1$ with label $a$
- $q_0$ to $q_2$ with label $a$

Transition table:

<table>
<thead>
<tr>
<th>State</th>
<th>Transition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$q_0$</td>
<td>${q_0, q_1}$</td>
</tr>
<tr>
<td>$q_1$</td>
<td>${q_2}$</td>
</tr>
<tr>
<td>$q_2$</td>
<td>${q_0}$</td>
</tr>
</tbody>
</table>
Examples of ambiguity in $\oplus$-NFAs

$k$-ambiguous $\oplus$-NFA

Lynette van Zijl  
Ambiguity in Symmetric Difference NFAs
Examples of ambiguity in ⊕-NFAs

Polynomially ambiguous ⊕-NFA

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$q_0$</td>
<td>$q_1$</td>
<td>$q_2$</td>
</tr>
<tr>
<td>$a$</td>
<td>$a$</td>
<td>$a$</td>
</tr>
</tbody>
</table>

- $q_0$: $\{q_1, q_2, q_3\}$
- $q_1$: $\{q_1, q_3\}$
- $q_2$: $\{q_2, q_3\}$
- $q_3$: $\{q_3\}$

Lynette van Zijl
Ambiguity in Symmetric Difference NFAs
Polynomially ambiguous $\oplus$-NFA

Lynette van Zijl  
Ambiguity in Symmetric Difference NFA
Examples of ambiguity in $\oplus$-NFAs

Exponentially ambiguous $\oplus$-NFA

Lynette van Zijl
Ambiguity in Symmetric Difference NFAs
Exponentially ambiguous $\oplus$-NFA

Lynette van Zijl
Ambiguity in Symmetric Difference NFAs
Ambiguity: current status

What have we achieved?
- Shown patterns to form ambiguous behaviour in structure

What remains?
- Succinct examples for each ambiguity class
- Families of languages to show relationship between ambiguity classes
- As above, but between traditional NFAs and ⊕-NFAs
Ambiguity: current status

What have we achieved?
- Shown patterns to form ambiguous behaviour in structure

What remains?
- Succinct examples for each ambiguity class
- Families of languages to show relationship between ambiguity classes
- As above, but between traditional NFAs and $\oplus$-NFAs
Ambiguity: current status

What have we achieved?
- Shown patterns to form ambiguous behaviour in structure

What remains?
- Succinct examples for each ambiguity class
- Families of languages to show relationship between ambiguity classes
- As above, but between traditional NFAs and $\oplus$-NFAs
Ambiguity: current status

What have we achieved?
- Shown patterns to form ambiguous behaviour in structure

What remains?
- Succinct examples for each ambiguity class
- Families of languages to show relationship between ambiguity classes
- As above, but between traditional NFAs and $\oplus$-NFAs
### Ambiguity: current status

**What have we achieved?**
- Shown patterns to form ambiguous behaviour in structure

**What remains?**
- Succinct examples for each ambiguity class
- Families of languages to show relationship between ambiguity classes
- As above, but between traditional NFAs and $\oplus$-NFAs
Questions?

Contact us
lvzijl@gmail.com
http://www.cs.sun.ac.za/lynnette