Two-way Two-tape Automata

Léo Exibard[1]

Join work with Olivier Carton[2] and Olivier Serre[2]

[1]Département d'Informatique
École Normale Supérieure de Lyon
France

[2]Institut de Recherche en Informatique Fondamentale
Université Paris Diderot - Paris 7
France

August 9, 2017

Outline

Setting

Automata

	Det	NDet	Alt
1-way	Reg		
2-way			

Transducers

Input	Output	Model	Example
1-way	1-way	Rat	$a_1 \dots a_n \mapsto a_2 \dots a_n a_1$
2-way	1-way	MSOt (det)	$w \mapsto w\widetilde{w}$
2-way	2-way		$w \mapsto w$ if $\lvert w \rvert = 2^n$

Setting

Automata

	Det	NDet	Alt
1-way	Reg		
2-way			

Transducers

Input	Output	Model	Example
1-way	1-way	Rat	$a_1 \dots a_n \mapsto a_2 \dots a_n a_1$
2-way	1-way	MSOt (det)	$w \mapsto w\widetilde{w}$
2-way	2-way		$w \mapsto w$ if $\lvert w \rvert = 2^n$

this talk

The model

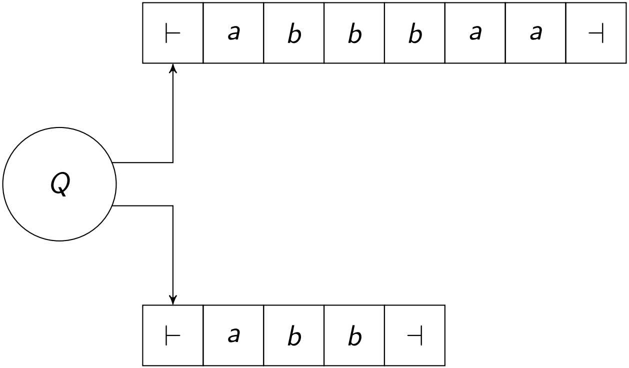


Figure: Recognizing $\mathcal{R} = \{(uwv, \widetilde{w} \mid u, v, w \in \Sigma^*)\}$

The model

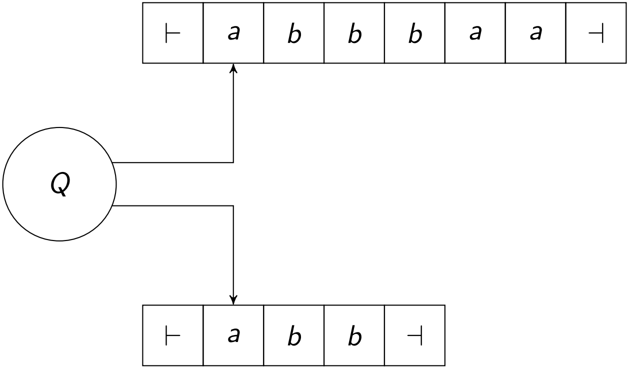


Figure: Recognizing $\mathcal{R} = \{(uwv, \widetilde{w} \mid u, v, w \in \Sigma^*)\}$

The model

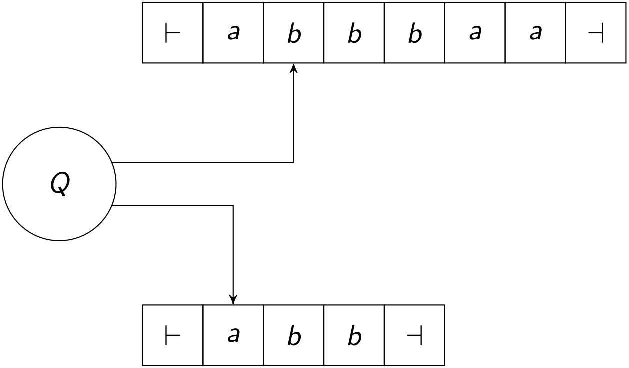


Figure: Recognizing $\mathcal{R} = \{(uwv, \widetilde{w} \mid u, v, w \in \Sigma^*)\}$

The model

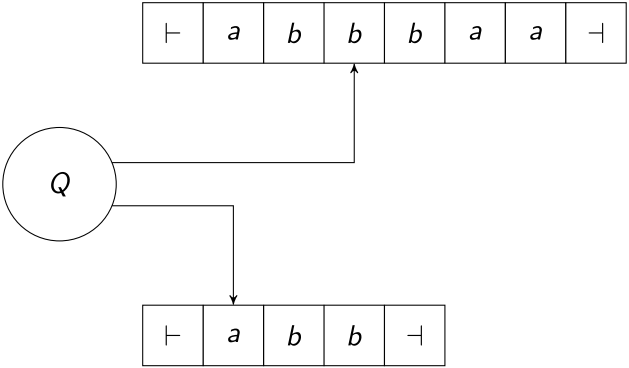


Figure: Recognizing $\mathcal{R} = \{(uwv, \widetilde{w} \mid u, v, w \in \Sigma^*)\}$

The model

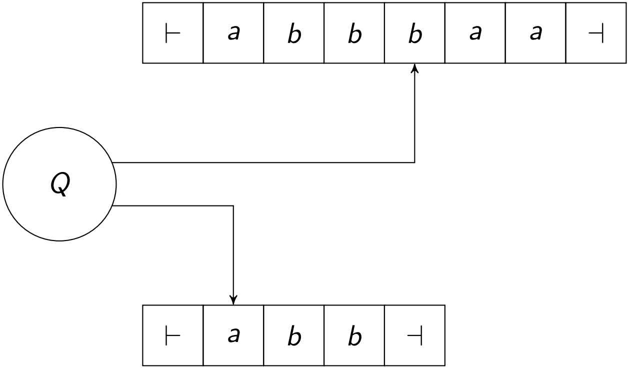


Figure: Recognizing $\mathcal{R} = \{(uwv, \widetilde{w} \mid u, v, w \in \Sigma^*)\}$

The model

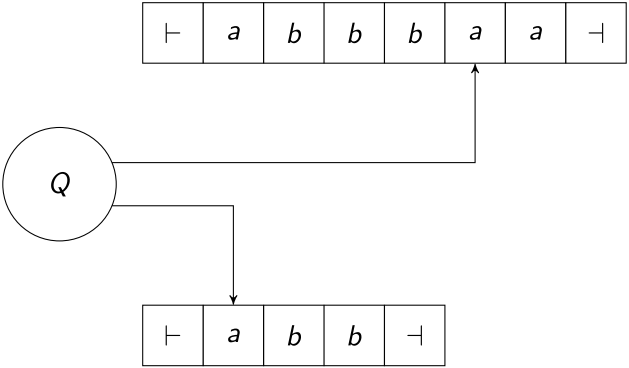


Figure: Recognizing $\mathcal{R} = \{(uwv, \widetilde{w} \mid u, v, w \in \Sigma^*)\}$

The model

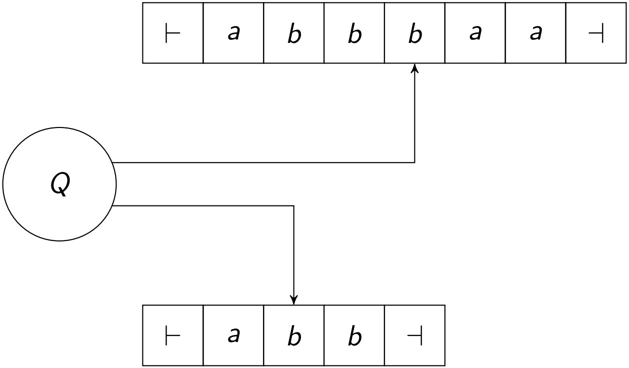


Figure: Recognizing $\mathcal{R} = \{(uwv, \widetilde{w} \mid u, v, w \in \Sigma^*)\}$

The model

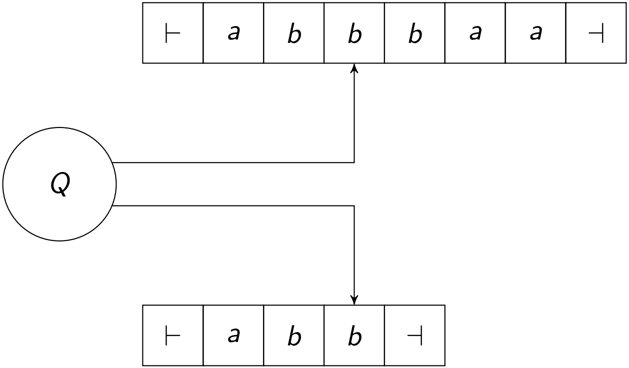


Figure: Recognizing $\mathcal{R} = \{(uwv, \widetilde{w} \mid u, v, w \in \Sigma^*)\}$

The model

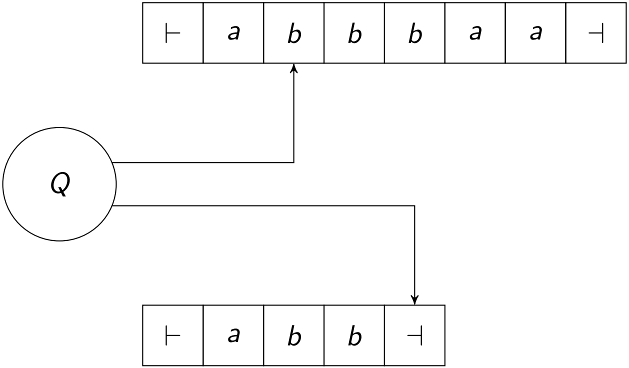
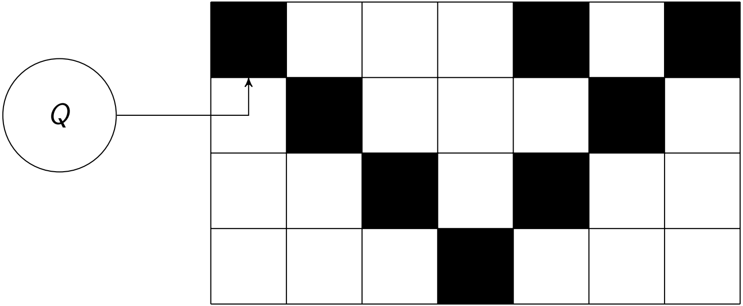
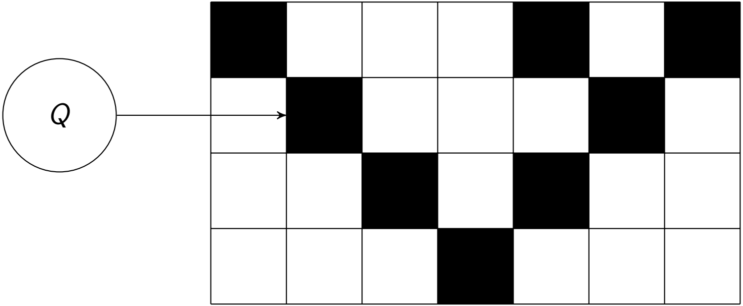


Figure: Recognizing $\mathcal{R} = \{(uwv, \widetilde{w} \mid u, v, w \in \Sigma^*)\}$

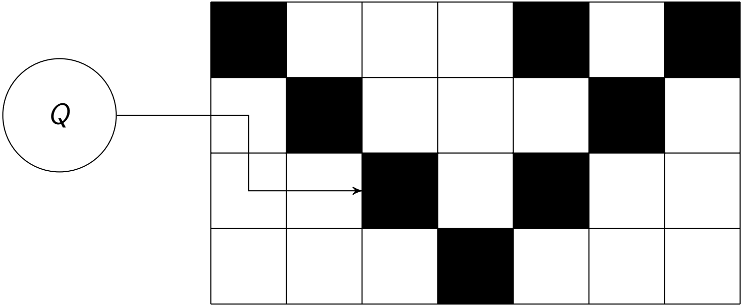
Picture-walking automata



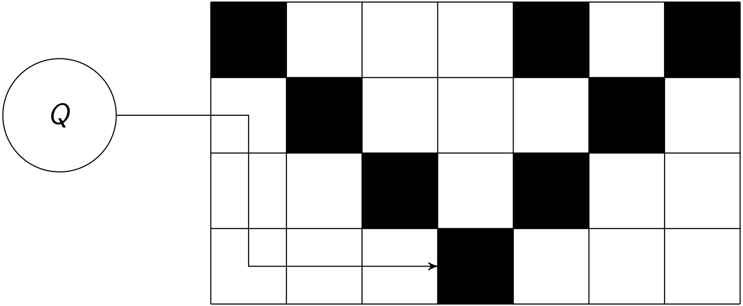
Picture-walking automata



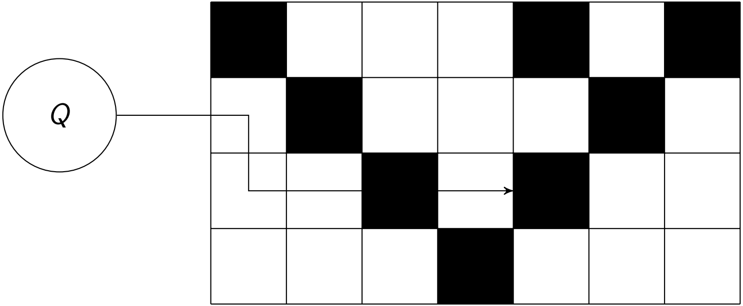
Picture-walking automata



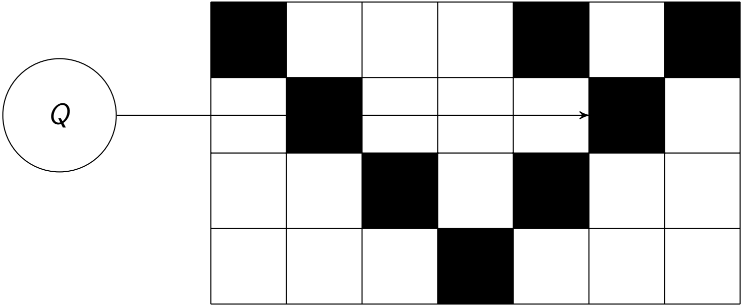
Picture-walking automata



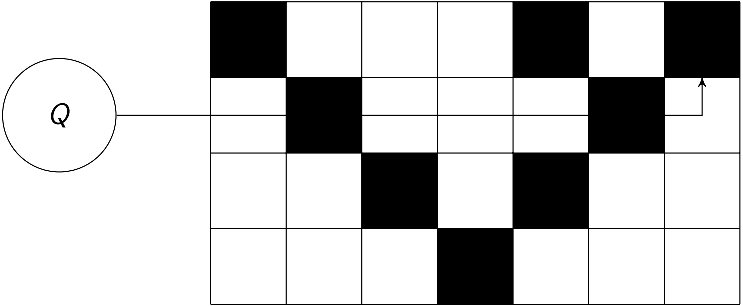
Picture-walking automata



Picture-walking automata



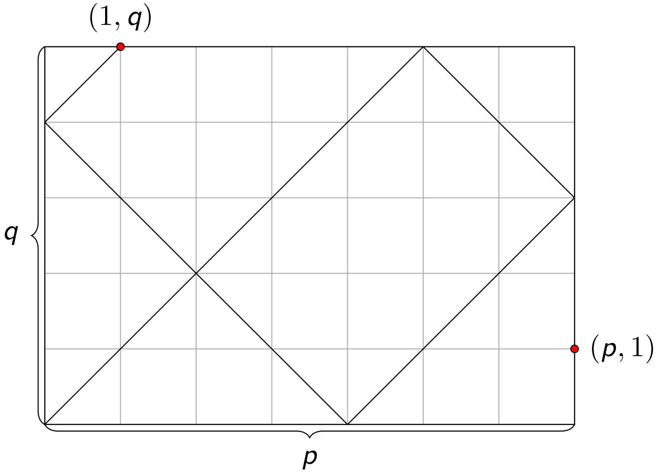
Picture-walking automata



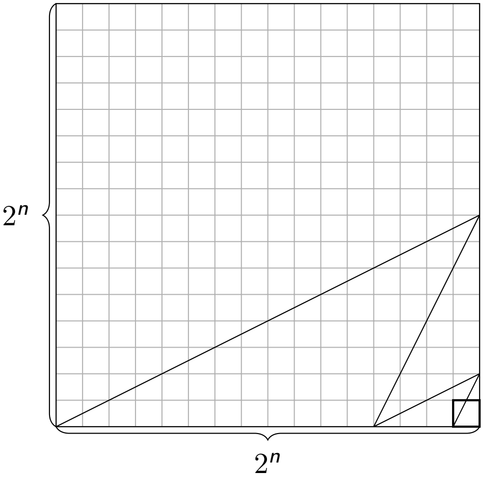
Bidimensional view

	a	b	b	b	a	a
a	(a,a)	(b,a)	(b,a)	(b,a)	(a,a)	(a,a)
b	(a,b)	(b,b)	(b,b)	(b,b)	(a,b)	(a,b)
b	(a,b)	(b,b)	(b,b)	(b,b)	(a,b)	(a,b)

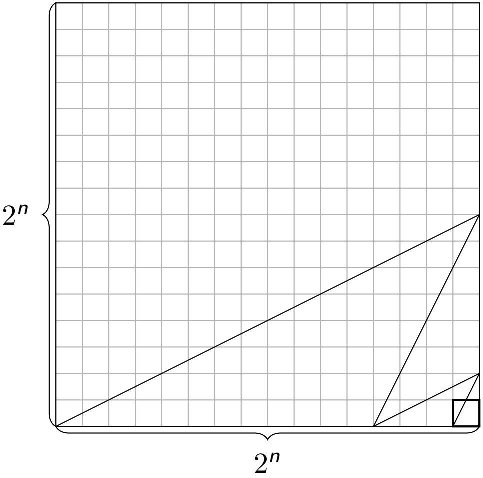
Coprime



Exponential square



Exponential square



- $\left\{2^{2^n}, 2^{2^n} \mid n \in \mathbb{N}\right\}$

Emptiness problem

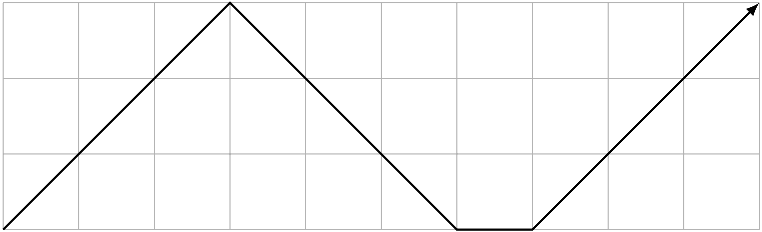
Proposition

The emptiness problem for 2-way 2-tape automata is undecidable, even for a unary alphabet.

- Binary case: PCP
- Unary case: Counter machines

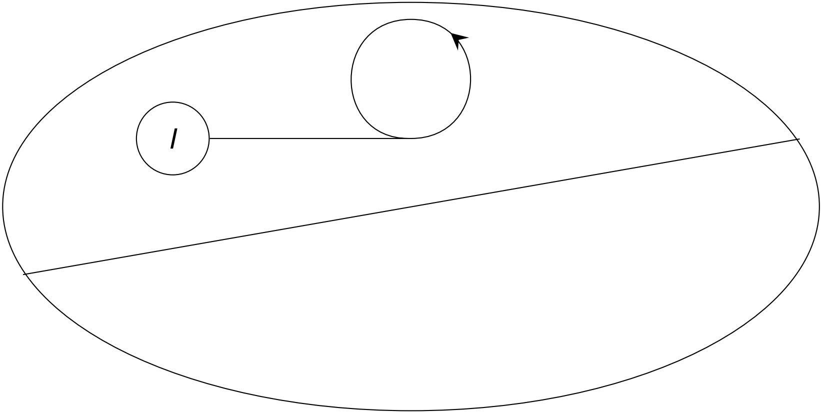
Non-determinism *vs determinism*

- Proof of Kari and Salo
- $L_{\text{billiard}} = \{(a^k, a^l) \mid \exists m, n \in \mathbb{N}, k = ml + n(l+1)\}$



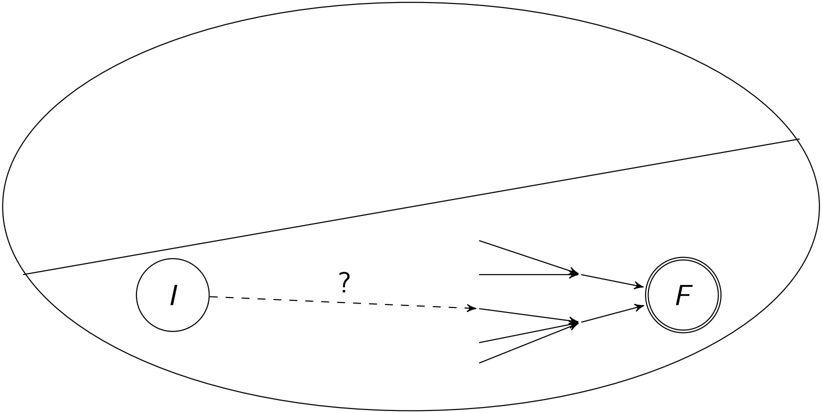
Removing cycles in computations

Fixed input (u, v)

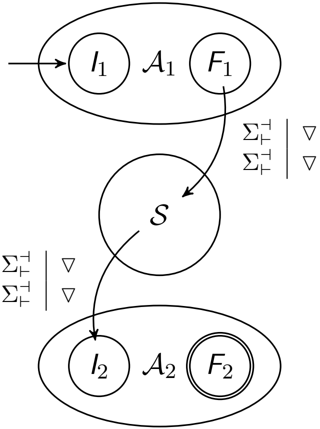


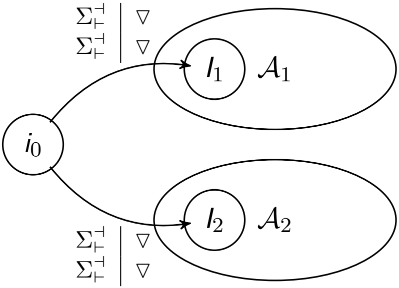
Removing cycles in computations

Fixed input (u, v)

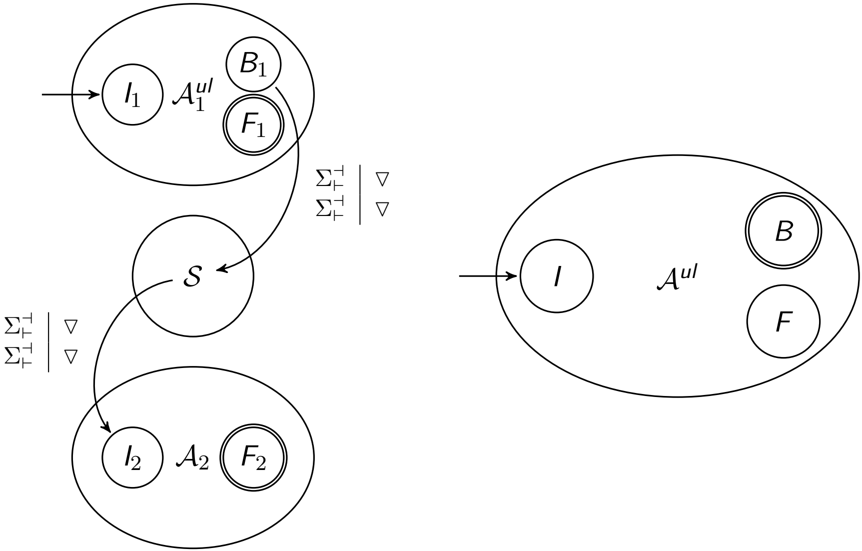


Intersection and union





Deterministic union and complementation



Summary

	$\cup$	$\cap$	^{c}
DFA	✓	✓	✓
NFA	✓	✓	✗

Composition

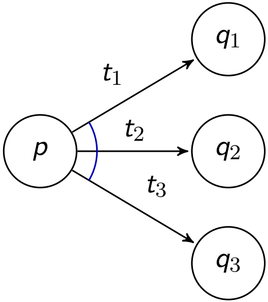
- Exponential bound on recognizable functions
- $\mathcal{E}$: $\begin{array}{l} 0\#1 \\ 000\#100\#010\#110\#001\#101\#011\#111 \end{array}$
- $\mathcal{E}$ is recognizable and exponential
- $\mathcal{E} \circ \mathcal{E}$ is not recognizable

Alternation

Definition

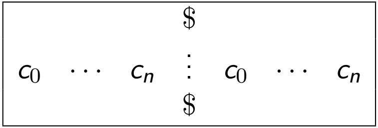
- $Q = Q_\forall \uplus Q_\exists$
- Computation: tree

Example (Non-deterministic *vs.* universal automaton)



Complementation: the alternation case

Theorem
Alternating 2-way 2-tape automata on a binary alphabet are not closed under complementation.

- Adaptation of a proof of Kari and Moore
- 

- $\mathcal{R} = \{(a^n, c_0\#\cdots\#c_n\$c_0\#\cdots\#c_n) \mid (c_0, \ldots, c_n) \text{ perm.}\}$
- $\mathcal{R} \in \mathrm{AFA}$
- $\mathcal{R} \notin \text{co-}\mathrm{AFA}$

Summary

	$\cup$	$\cap$	^{c}	$\circ$
DFA	✓	✓	✓	✗
NFA	✓	✓	✗	✗
AFA	✓	✓	✗	✗

Perspectives

- Are unary alternating automata closed under complementation?
- Links with counter machines
- Characterization of recognizable relations
- General framework for additional tapes

Complementation: the alternation case

Theorem

Alternating 2-way 2-tape automata on a binary alphabet are not closed under complementation.

- Adaptation of a proof of Kari and Moore
- $$\boxed{\begin{array}{ccccccc} & & & \$ & & & \\ c_0 & \cdots & c_n & \vdots & c_0 & \cdots & c_n \\ & & & \$ & & & \end{array}}$$
- $\mathcal{R} = \{(a^n, c_0\#\cdots\#c_n\$c_0\#\cdots\#c_n) \mid (c_0, \ldots, c_n) \text{ perm.}\}$

Complementation: the alternation case

Representing permutations

0	0	1	\$	0	0	1
1	0	0	\$	1	0	0
0	1	0	\$	0	1	0

$$(a^3, 010\#001\#100\$010\#001\#100)$$

Complementation: the alternation case

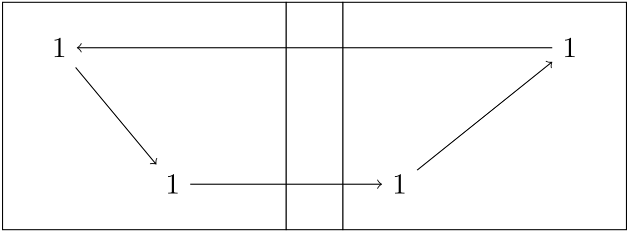
Theorem

Alternating 2-way 2-tape automata on a binary alphabet are not closed under complementation.

- Adaptation of a proof of Kari and Moore
- $$\boxed{\begin{array}{ccccccc} & & & \$ & & & \\ c_0 & \cdots & c_n & \vdots & c_0 & \cdots & c_n \\ & & & \$ & & & \end{array}}$$
- $\mathcal{R} = \{(a^n, c_0\# \cdots \# c_n \$ c_0 \# \cdots \# c_n) \mid (c_0, \ldots, c_n) \text{ perm.}\}$
- $\mathcal{R} \in \mathrm{AFA}$

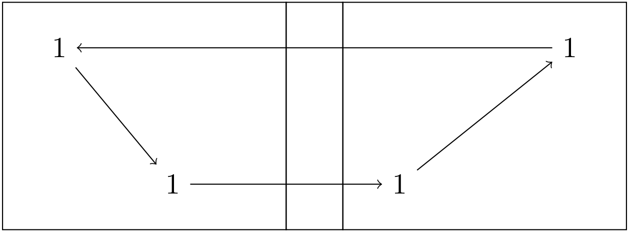
Complementation: the alternation case

Detecting inversions



Complementation: the alternation case

Detecting inversions



- Can be generalized to other picture languages

Complementation: the alternation case

Theorem

Alternating 2-way 2-tape automata on a binary alphabet are not closed under complementation.

- Adaptation of a proof of Kari and Moore
- $$\begin{array}{|ccccccc|} & & & \$ & & & \\ c_0 & \cdots & c_n & \vdots & c_0 & \cdots & c_n \\ & & & \$ & & & \\ \end{array}$$
- $\mathcal{R} = \{(a^n, c_0\# \cdots \#c_n\$c_0\# \cdots \#c_n) \mid (c_0, \ldots, c_n) \text{ perm.}\}$
- $\mathcal{R} \in \mathrm{AFA}$
- $\mathcal{R} \notin \mathrm{co\text{-}AFA}$

Complementation: the alternation case

Communication complexity

$$(a^n, c_0 \# \cdots \# c_n \$ c'_0 \# \cdots \# c'_n)$$

$$c_0 \dots c_n \stackrel{?}{=} c'_0 \dots c'_n$$

Complementation: the alternation case

Game-theoretic proof

Computation rejected when $\exists\mathcal{C}$ s.t.:

(i) $i \in \mathcal{C}$

(ii) $c \in \mathcal{C}$ existential $\Rightarrow$ every successor is in $\mathcal{C}$

(iii) $c \in \mathcal{C}$ universal $\Rightarrow$ a successor is in $\mathcal{C}$

(iv) $\mathcal{C} \cap F = \emptyset$

$\rightarrow$ Winning strategy for the universal player

Complementation: the alternation case

Combinatorial argument

- Assume there exists $\mathcal{A}$ recognizing $\overline{\mathcal{R}}$
- Strategy = annotated tapes
- Local annotations
- Counting argument: there exists σ, σ' coiciding around the $\$$
- Glue strategies to reject $\sigma\$\sigma'$