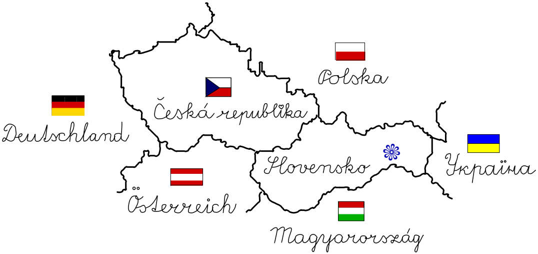
State Complexity of Union and Intersection of Binary Suffix-Free Languages

Galina Jirásková and Pavol Olejár

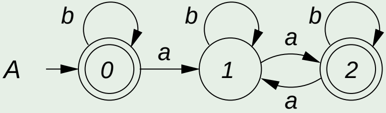
Slovak Academy of Sciences and Šafárik University, Košice



Outline

- DFAs and NFAs
- Minimal Automata
- Suffix−Free Languages
- Descriptional Complexity of Intersection and Union
- Summary and Open Problems

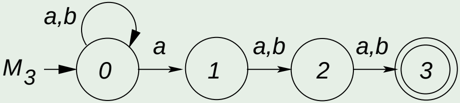
Example (DFA $A = (Q, \Sigma, \delta, s, F)$)



- $Q = \{0, 1, 2\}$
- $\Sigma = \{a, b\}$
- $\delta : Q \times \Sigma \to Q$
- $s = 0$
- $F = \{0, 2\}$
  (complete)

$L(A) = \{w \in \{a, b\}^* \mid$ the number of a's in w is even$\}$

Example (NFA $M_3 = (Q, \Sigma, \delta, s, F)$)



- $Q = \{0, 1, 2, 3\}$
- $\Sigma = \{a, b\}$
- $\delta : Q \times \Sigma \to 2^Q$
- $s = 0$
- $F = \{3\}$
  (ε-free, single initial state)

$L_3 = \{w \in \{a, b\}^* \mid w$ contains an a in the third position from the end$\}$

Minimal Deterministic Finite Automata

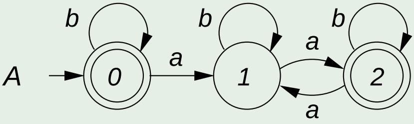
Definition

Automata A and B are equivalent
if $L(A) = L(B)$.

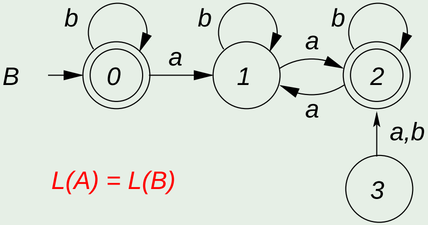
Definition

A dfa A is minimal
if every equivalent dfa
has at least as many states as A.

Example



$$L(A) = \{w \in \{a, b\}^* \mid \#_a(w) \text{ is even}\}$$



Minimal Deterministic Finite Automata

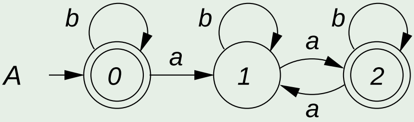
Definition

Automata A and B are equivalent if $L(A) = L(B)$.

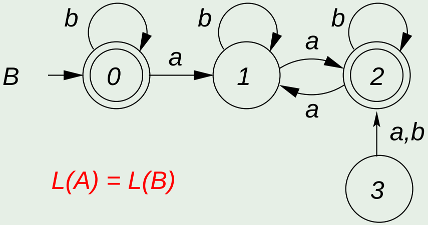
Definition

A dfa A is minimal if every equivalent dfa has at least as many states as A.

Example



$$L(A) = \{w \in \{a,b\}^* \mid \#_a(w) \text{ is even}\}$$



L(A) = L(B)

Minimal Deterministic Finite Automata

Definition

Automata A and B are equivalent if $L(A) = L(B)$.

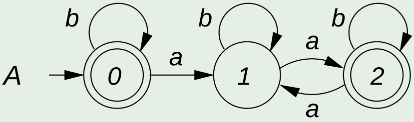
Definition

A dfa A is minimal
if every equivalent dfa
has at least as many states as A.

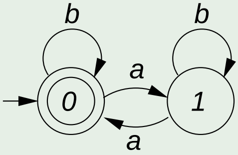
Theorem

A dfa is minimal if
(1) each state is reachable,
(2) no two states are equivalent.

Example



$L(A) = \{w \in \{a, b\}^* \mid \#_a(w) \text{ is even}\}$



The minimal dfa accepting $L(A)$.

Minimal Deterministic Finite Automata

Definition

Automata A and B are equivalent if $L(A) = L(B)$.

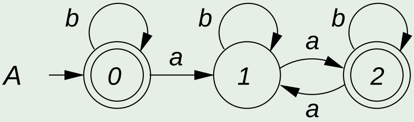
Definition

A dfa A is minimal if every equivalent dfa has at least as many states as A.

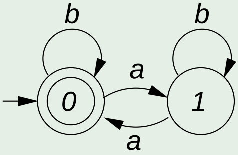
Definition

The state complexity of L is the number of states in the minimal DFA for L.

Example



$$L(A) = \{w \in \{a, b\}^* \mid \#_a(w) \text{ is even}\}$$



The state complexity of $L(A)$ is 2.

Fooling-Set Lower-Bound Method

Definition

A set of pairs of strings $\{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$ is a fooling set for L, if

(F1) $x_i y_i \in L$ for all i,

(F2) if $i \neq j$,
then $x_i y_j \notin L$ or $x_j y_i \notin L$.

Fooling-Set Lower-Bound Method

Definition

A set of pairs of strings $\{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$ is a fooling set for L, if

(F1) $x_i y_i \in L$ for all i,

(F2) if $i \neq j$, then $x_i y_j \notin L$ or $x_j y_i \notin L$.

Example

$\Sigma = \{a,c,m,n\}$

$L = \text{nc}(\text{m}+\text{a})^*$

$$\mathcal{A} = \{(\varepsilon, ncma), (\text{n}, \text{cma}), (\text{nc}, \text{ma})\}$$

is a fooling set for L.

Fooling-Set Lower-Bound Method

Definition

A set of pairs of strings $\{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$ is a fooling set for L, if

(F1) $x_i y_i \in L$ for all i,

(F2) if $i \neq j$, then $x_i y_j \notin L$ or $x_j y_i \notin L$.

Lemma (Aho 83, Birget 92)

If $\mathcal{A}$ is a fooling set for L, then every NFA accepting L has at least $|\mathcal{A}|$ states.

Example

$\Sigma = \{a,c,m,n\}$

$L = nc(m+a)^*$

$\mathcal{A} = \{(\varepsilon, ncma), (n, cma), (nc, ma)\}$

is a fooling set for L.

The nondeterministic state complexity of L is 3.

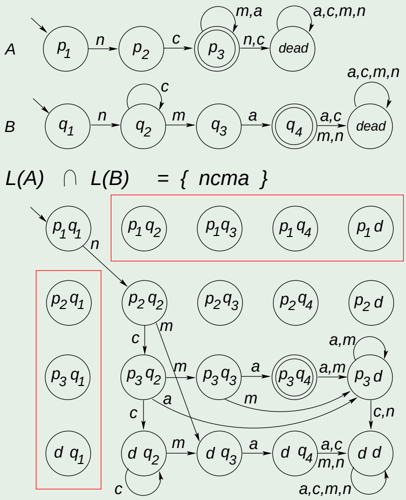
Cross-Product Automaton for Intersection and Union

Given DFAs

$A = (Q_A, \Sigma, \delta_A, s_A, F_A)$
$B = (Q_B, \Sigma, \delta_B, s_B, F_B)$
their cross-product automaton is $(Q, \Sigma, \delta, s, F)$ with

- $Q = Q_A \times Q_B$
- $\delta((p, q), a) = (\delta_A(p, a), \delta_B(q, a))$
- $s = (s_A, s_B)$
- $F = F_A \times F_B$ for intersection
- $F = (F_A \times Q_B) \cup (Q_A \times F_B)$ for union

Example ($\Sigma = \{a, c, m, n\}$)



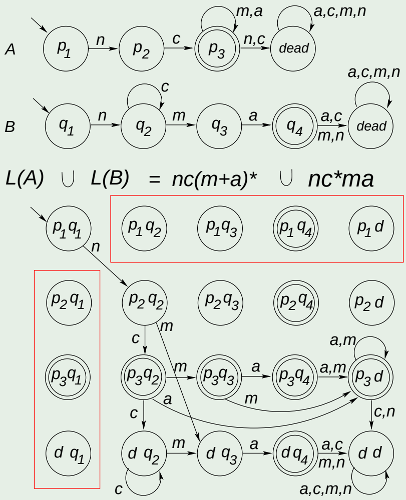
Cross-Product Automaton for Intersection and Union

Given DFAs

$A = (Q_A, \Sigma, \delta_A, s_A, F_A)$
$B = (Q_B, \Sigma, \delta_B, s_B, F_B)$
their cross-product automaton is $(Q, \Sigma, \delta, s, F)$ with

- $Q = Q_A \times Q_B$
- $\delta((p, q), a) = (\delta_A(p, a), \delta_B(q, a))$
- $s = (s_A, s_B)$
- $F = F_A \times F_B$ for intersection
- $F = (F_A \times Q_B) \cup (Q_A \times F_B)$ for union

Example ($\Sigma = \{a, c, m, n\}$)



Outline

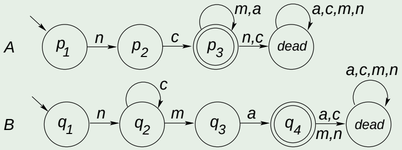
- Suffix−Free Languages
- Descriptional Complexity of Intersection and Union
- Summary and Open Problems

Suffix-Free Regular Languages

Definition

A regular language L is suffix-free if there are no two different strings u and v in L such that $u = wv$.

Example



$L(A) = nc(m+a)^*$

$L(B) = nc^*ma$

Theorem (Han, Salomaa 2008)

In the minimal (complete) DFA for a suffix free language,

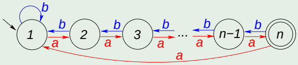
- *there is a dead state;*
- *no transition goes to the initial state.*

State Complexity of Operations on Suffix-Free Languages (Han, Salomaa 2008)

Operation	Suffix-Free	Σ	Regular	Σ
L^*	$2^{n-2}+1$	4	$2^{n-1}+2^{n-2}$	2
L^R	$2^{n-2}+1$	3	2^n	2
$K \cdot L$	$(m-1)2^{n-2}+1$	4	$(m-1)2^n+2^{n-1}$	2
$K \cap L$	$mn-2(m+n)+6$	3	mn	2
$K \cup L$	$mn-(m+n)+2$	5	mn	2

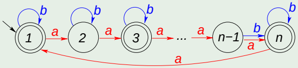
Binary Worst-Case Examples

Example (Star, Maslov 1970)



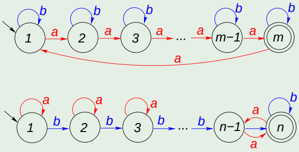
$$2^{n-1} + 2^{n-2}$$

Example (Reversal, Leiss 1981)



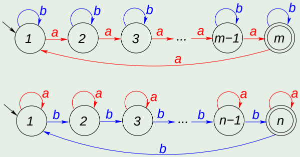
$$2^n$$

Example (Concatenation)



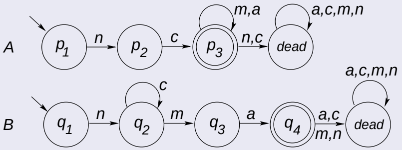
$$(m-1)2^n + 2^{n-1}$$

Example (Union, Maslov 1970)



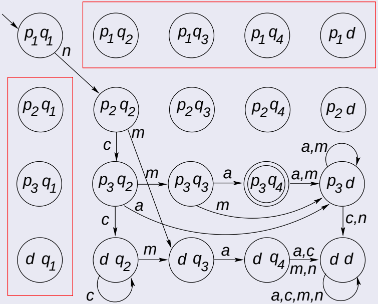
$$mn$$

Operation	Suffix-Free	Σ	Regular	Σ
L^*	$2^{n-2}+1$	4	$2^{n-1}+2^{n-2}$	2
L^R	$2^{n-2}+1$	3	2^n	2
$K \cdot L$	$(m-1)2^{n-2}+1$	4	$(m-1)2^n+2^{n-1}$	2
$K \cap L$	$mn-2(m+n)+6$	3	mn	2
$K \cup L$	$mn-(m+n)+2$	5	mn	2



Upper bounds on

L(A) ∩ L(B) and L(A) ∪ L(B)



Our Results on Binary Suffix-Free Regular Languages

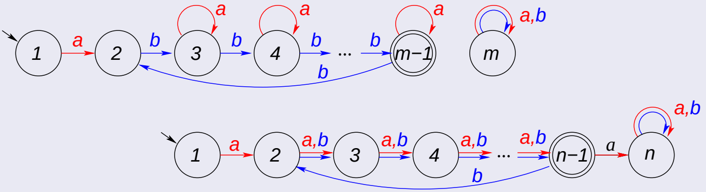
Theorem (State complexity of intersection)

The upper bound $mn - 2(m+n) + 6$ on the state complexity of intersection of suffix-free languages is tight in the binary case.

Theorem (State complexity of union)

The upper bound $mn - (m+n) + 2$ on the state complexity of union of suffix-free languages is tight in the binary case.

Proof. Witness languages for both operations:

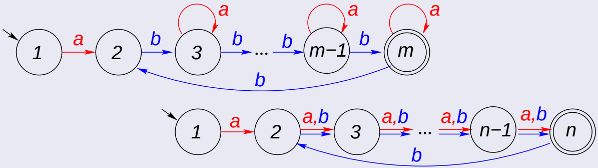


□

Our Results on Binary Suffix-Free Regular Languages II

Theorem (Nondeterministic state complexity of intersection)

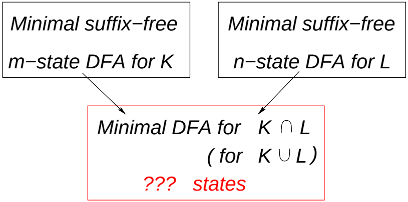
The nondeterministic state complexity of intersection of binary suffix-free languages is $mn - (m + n) + 2$.



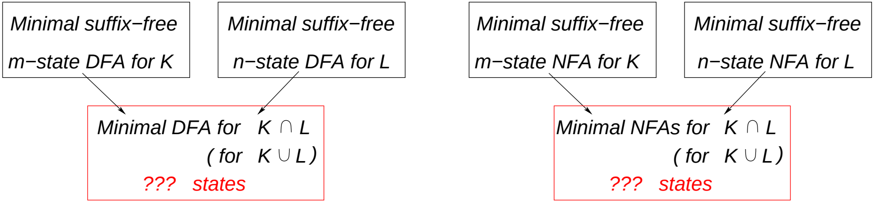
Theorem (Nondeterministic state complexity of union)

The nondeterministic state complexity of union of binary suffix-free languages is $m + n - 1$.

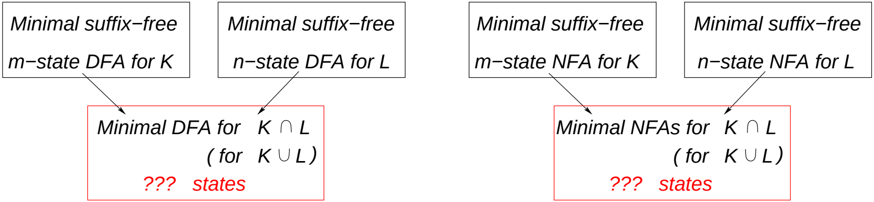
Intersection and Union of Suffix-Free Languages: Not Only Worst Cases



Intersection and Union of Suffix-Free Languages: Not Only Worst Cases



Intersection and Union of Suffix-Free Languages: Not Only Worst Cases



Theorem (Deterministic case)

The size of minimal DFA for $K \cap L$ *may be* $1, 2, \ldots, mn - 2(m+n) + 6$ *for a ternary alphabet.*

The size of minimal DFA for $K \cup L$ *may be* $1, 2, \ldots, mn - 2(m+n) + 6$ *for a ternary alphabet.*

Theorem (Nondeterministic case)

The size of minimal NFA for $K \cap L$ *may be* $1, 2, \ldots, mn - (m+n) + 2$ *for a four-letter alphabet.*

The size of minimal NFA for $K \cup L$ *may be* $2, 3, \ldots, m + n - 1$ *for a ternary alphabet.*

Outline

Summary and Open Problems

Summary and Open Problems

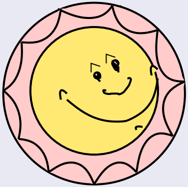
Our Results on Binary Suffix-Free Languages:

	Binary Suffix-Free	Binary Regular
$\mathrm{sc}(\textit{intersection})$	$mn - 2(m+n) + 6$	mn
$\mathrm{sc}(\textit{union})$	$mn - (m+n) + 2$	mn
$\mathrm{nsc}(\textit{intersection})$	$mn - (m+n) + 2$	mn
$\mathrm{nsc}(\textit{union})$	$m + n - 1$	$m + n + 1$

Are There Any Holes?

	sc	Σ	nsc	Σ
$K \cap L$	$1..mn - 2(m+n) + 6$	3	$1..mn - (m+n) + 2$	4
$K \cup L$	$1..mn - 2(m+n) + 6$	3	$2..m + n - 1$	3

Thank You for Your Attention



Dziękuję za uwagę!