Finite Transducers and Nondeterministic State Complexity

Grigoriy Povarov

Ural State University

Russian-Indian Workshop
“Algebra, Combinatorics, Complexity”
Ekaterinburg, Russia, 2008

Outline

Common problem

Language L

Operation τ

Complexity (measure)

Problem

Find the complexity of $\tau(L)$ in terms of the complexity of L and the complexity of τ

Common problem: our case

Language L – regular language

Operation τ – finite transducer

Complexity (measure) – nondeterministic state complexity

Problem

Find the complexity of $\tau(L)$ in terms of the complexity of L and the complexity of τ

Nondeterministic finite automata

NFA $(Q, \Sigma, \delta, q_0, F)$

- Q – finite set of states
- Σ – finite set of letters
- $\delta \subset Q \times \Sigma \times Q$ – set of transitions
- $q_0 \in Q$ – initial state
- $F \subset Q$ – set of final states

Nondeterministic state complexity

Regular language L

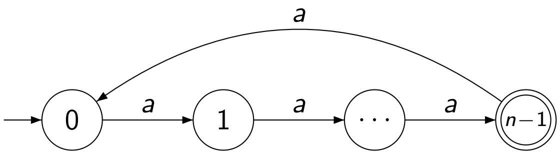
NFA A: $L(A) = L$

Minimal number of states – nondeterministic state complexity of L

$\mathrm{nsc}(L)$

Nondeterministic state complexity: example

$$L_n = \{a^k \mid k \equiv n-1 \pmod{n}\}$$



$\mathrm{nsc}(L_n) = n$

Nondeterministic state complexity: known results

M. Kutrib, M. Holzer, *State Complexity of Basic Operations on Nondeterministic Finite Automata* (2003)

operation	*nsc*	*sc*
$\cup$	$m+n+1$	mn
$\cap$	mn	mn
C	$O(2^{n-1})$	n
R	$n+1$	2^n
$\cdot$	$m+n$	$(2m-1)2^{n-1}$
$*$	$n+1$	$3 \cdot 2^{n-2}$

Finite transducer: definition

Finite transducer $(Q, \Sigma, \Delta, \delta, q_0, F)$

- Q – finite set of states
- Σ – finite set of input letters
- Σ – finite set of output letters
- $\delta \subset Q \times \Sigma^* \times \Delta^* \times Q$ – finite set of transitions
- $q_0 \in Q$ – initial state
- $F \subset Q$ – set of final states

Finite transducer: language transformation

$$\mathcal{R}(\tau) = \{(u, v) \in \Sigma^* \times \Delta^* \mid \exists q \in F : q_0 \overset{u,v}{\twoheadrightarrow} q\}$$

$$\tau(L) = \{v \in \Delta^* \mid \exists u \in L : (u, v) \in \mathcal{R}(\tau)\}$$

Lemma (see J.Sakarovitch, *Éléments de théorie des automates*)

For finite transducer τ and regular language L the language $\tau(L)$ is regular.

Hamming distance: recall

Hamming distance between two words of equal length – number of positions where the words are different

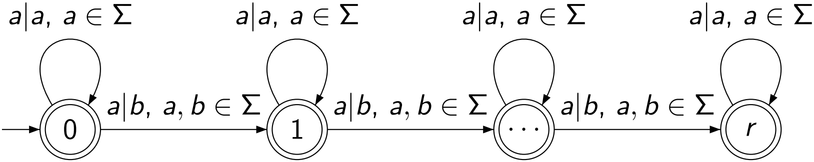
"toned" and "roses": the distance is 3

Hamming neighborhood of radius r of a language $L \subset \Sigma^*$ (r-neighborhood)

$\mathcal{O}(L, r) = \{w \in \Sigma^* \mid \exists u \in L \mid h(w, u) \leq r\}$.

Finite transducer: example

Finite transducer τ_r for Hamming r-neighborhood ($\Sigma = \{a, b\}$)



Normalized finite transducer

If $\delta \subset Q \times (\Sigma \cup \{\lambda\}) \times (\Delta \cup \{\lambda\}) \times Q$ then the transducer τ is normalized.

The transducer τ_r for Hamming neighborhood is normalized.

Lemma (see J.Karhumäki, *Automata and Formal Languages*)

For any finite transducer there is an equivalent normalized finite transducer.

Main result

Theorem

If L is a regular language, τ is a normalized finite transducer then

$$\mathrm{nsc}(\tau(L)) \leq |\tau|\,\mathrm{nsc}(L).$$

This bound is tight: for any $r > 1$ and $n > r + 1$ there exist regular language L and normalized finite transducer τ and

$$\mathrm{nsc}(L) = n, |\tau| = r, |\,\mathrm{nsc}(\tau(L)) = nr.$$

Proof of the upper bound: idea

The NFA of $\tau(L)$ is the “cartesian product” transducer τ by NFA of L.

The reading image-word in NFA $\tau(L)$ corresponds synchronized reading the original-word in the NFA of L and reading the pair (original-word, image-word) in the transducer τ.

Proof of the upper bound: format definition

L – regular language

$\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$ – NFA of L with minimal number of states

$\tau = (P, \Sigma, \Delta, \gamma, p_0, E)$ – normalized finite transducer

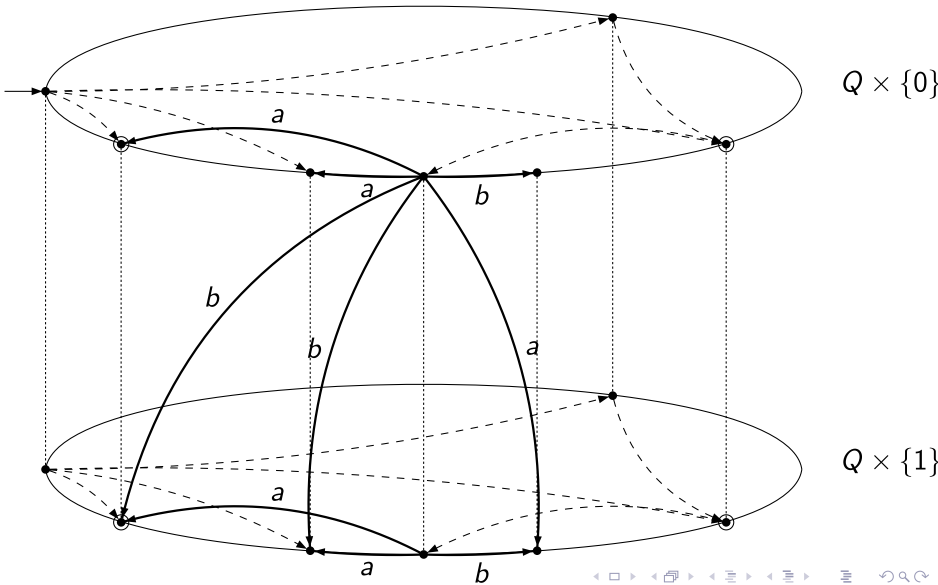
$$\mathcal{B} = (Q \times P, \Delta, \epsilon, (q_0, p_0), F \times E)$$

where

$$\epsilon = \left\{ ((q,p), b, (q', p')) \mid \exists a \in \Sigma \cup \{\lambda\} : p \xrightarrow{a|b} p' \text{ and } q \xrightarrow{a} q' \right\}$$

for $q, q' \in Q,\ p, p' \in P,\ b \in \Delta \cup \{\lambda\}$

Proof of the upper bound: illustration for τ_2



Upper bound is tight

$L_n = \{a^k \mid k \equiv n-1 (\mod n)\}$, $\text{nsc}(L_n) = n$

τ_r – normalized finite transducer for Hamming r-neighborhood, $|\tau_r| = r+1$

$\text{nsc}(\tau_r(L_n)) = |\tau_r| \, \text{nsc}(L)$ (for $n > r$)

What about deterministic state complexity?

L – regular language

τ – normalized finite transducer

$sc(\tau(L)) \leq 2^{sc(L)|\tau|}$ – straightforward subset-construction

Theorem (G.Povarov, *Descriptive Complexity of the Hamming Neighborhood of a Regular Language* (2007))

K_n – regular language, $sc(K_n) = n$, $n > 4$

$sc(\mathcal{O}(K_n, 1)) = \frac{3}{8} n \cdot 2^n - 2^{n-4} + n$

Thank you!