Complementing Unary Nondeterministic Automata

Filippo Mera and Giovanni Pighizzini
Dipartimento di Informatica e Comunicazione
Università degli Studi di Milano, Italy
**PROBLEM**
Comparing the number of states needed to accept a language and its complement, i.e.,

Given an \( n \)-state automaton accepting \( \mathcal{L} \), how many states are necessary and sufficient to accept \( \mathcal{L}^C \)?

**Deterministic Automata**
Trivial: it suffices to complement the set of final states.

**Nondeterministic Automata**
Upper bound: \( 2^n \) (just convert to a deterministic automaton).
This bound cannot be improved [Next talk].
Present paper

Investigation of this problem for the unary case.

Main result:

\( L \) is accepted by a “small” nfa

\[ \downarrow \]

each nfa accepting \( L^C \) must be “large”
What do “small” and “large” mean?

**Small:** The automaton and the language it accepts are witnesses of the gap between nfa’s and dfa’s, i.e., the nfa has $n$-states and each dfa accepting the same language has at least $e^{\Theta(\sqrt{n \ln n})}$ states.

**Large:** The automaton has at least as many states as a deterministic automaton (nondeterminism is thus useless.)
Unary Deterministic Automata

Size of an automaton $A \equiv (\lambda, \mu)$

Any unary regular language is ultimately cyclic. It is cyclic for words of length $\geq \mu$, being $(\lambda, \mu)$ the size of a dfa accepting it.

If $A$ is minimum then $\lambda$ is the *ultimate period* of the language considered.
Theorem [Chr86]:

Any nfa with $n$ states can be simulated with an nfa in Chrobak Normal Form having size at most $\left(n, O\left(n^2\right)\right)$.
State complexity of a regular language:

\[ sc(L) \equiv \text{number of states of the smallest deterministic automaton accepting it.} \]

Nondeterministic state complexity:

\[ nsc(L) \equiv \text{number of states of a minimal nondeterministic automaton accepting it.} \]

Theorem [Jiang, McDowell, Ravikumar 91]

If \( \lambda \) is the ultimate period of \( L \) and \( \lambda \) factorizes as \( p_1^{k_1} \cdot p_2^{k_2} \cdots p_s^{k_s} \), then we have \( nsc(L) \geq p_1^{k_1} + p_2^{k_2} + \cdots + p_s^{k_s} \).
Main result:

If a unary language $\mathcal{L}$ with ultimate period $\lambda = p_1^{k_1} \cdot p_2^{k_2} \cdot \ldots \cdot p_s^{k_s}$ is accepted by an nfa $A$ with $p_1^{k_1} + p_2^{k_2} + \ldots + p_s^{k_s}$ states in its cycles, then $\mathcal{L}^C$ requires at least $\lambda$ cyclic states.

Notes:

- $A$ has the smallest possible number of states with respect to the ultimate period of $\mathcal{L}$
- Nondeterminism does not allow to reduce the number of states for automata accepting $\mathcal{L}^C$
Proof (sketch)

W.l.o.g. \( A \) is in Chrobak Normal Form with cycles of lengths \( p_1^{k_1}, \ldots, p_s^{k_s} \)

Choose \( m_i \) and \( m \) such that:

- \( a^{m_i} \) is accepted in the \( i \)-th cycle,
- \( a^{m_i + p_i^{k_i-1}} \notin \mathcal{L} \), and
- \( \forall i \ m \equiv m_i + p_i^{k_i-1} \pmod{p_i^{k_i}} \) (Chinese Remainder Theorem).

Then:
- \( a^{m_i + p_i^{k_i}x} \in \mathcal{L} \), for \( i = 1, \ldots, s, \ x \geq 0 \),
- \( a^m \notin \mathcal{L} \).
Let $p$ be the length of a cycle of the automaton $A'$ accepting $L^C$ visited during an accepting computation on $a^m$.

Then:

- $A'$ accepts each $a^{m+py}$, with $y \geq 0$,
- $A'$ must reject each $a^{mi+p^ikix}$, $i = 1, \ldots, s$, $x \geq 0$.

Hence, for $i = 1, \ldots, s$, there are no integers $x, y \geq 0$ such that

$$m + py = mi + p^ikix,$$

But $m = mi + hp^{ki} + p^{ki-1}$, for some integer $h$.

Hence, there are no integers $x, y \geq 0$ such that

$$hp^{ki} + p^{ki-1} = p^{ki}x - py$$

This implies that $p^{ki} | p$, for $i = 1, \ldots, s$. 
Nonunary alphabets

Main Theorem cannot be extended to nonunary languages:

**Theorem**

A sequence $\mathcal{L}_n$ of languages can be exhibited such that:

- $\text{nsc} (\mathcal{L}_n) = n$
- $\text{sc} (\mathcal{L}_n) = 2^n$

($\mathcal{L}_n$ is thus a witness of the gap between nondeterministic and deterministic automata)

and

- $\text{nsc} (\mathcal{L}_n^C) \leq n + 1$
Question
What can be said about the converse of Main Theorem?

i.e.,

does the fact that each nfa accepting $L$ is “large” imply that $L^C$ has a “small” nfa?

The answer is negative:

Theorem
Let $p_1^{k_1} \cdot p_2^{k_2} \cdots p_s^{k_s}$ be the prime factorization for an arbitrary integer $\lambda$ and consider

$L_\lambda = \{a^m \mid \# \{i \mid p_i^{k_i} \text{ divides } m\} \text{ is even} \}$.

We have both $\text{nsc} (L_\lambda) = \lambda$ and $\text{nsc} (L_\lambda^C) = \lambda$.

The smallest nfa accepting $L_\lambda$ (or $L_\lambda^C$) is actually a dfa made of a single cycle of length $\lambda$. 
Sketch of the proof
in the case of $\lambda = p_1^{k_1} \cdots p_s^{k_s}$, with $s$ even.

We show that each nfa $A$ accepting the language
\[ L_\lambda = \{ a^m | \# \{ i | p_i^{k_i} \text{ divides } m \} \text{ is even} \} \]
contains one simple cycle of at least $\lambda$ states.

Consider the length $\ell$ of a simple cycle crossed in an accepting computation $C$ on an input $a^{\lambda H}$, for a sufficiently large $H$.

Let $m_j = H\lambda + \ell \frac{\lambda}{p_j^{k_j}}$, for $j = 1, \ldots, s$.

By “pumping” the computation $C$ with the cycle of length $\ell$, we get that $a^{m_j} \in L_\lambda$.

Since each $p_i^{k_i}$, with $i \neq j$, divides $m_j$, this implies that even $p_j^{k_j}$ must divide $m_j$.

Hence, we can easily conclude that each $p_j^{k_j}$ divides $\ell$ and, finally, that $\lambda$ divides $\ell$. 
Conclusions and open problems

We have been studying the problem in its “extreme” case, investigating languages that witness the gap considered.

Problem:

Bounds should be found, that simultaneously apply to \( \text{nsc}(L) \) and \( \text{nsc}(L^C) \) for unary languages.

In other words, it is of some interest to investigate the trade-off between the nondeterministic complexity of unary languages and that of their complements.