A Survey on Fooling Sets as Effective Tools for Lower Bounds on Nondeterministic Complexity

Michal Hospodár[1], Galina Jirásková[1(✉)], and Peter Mlynárčik[1,2]

[1] Mathematical Institute, Slovak Academy of Sciences,
Grešákova 6, 040 01 Košice, Slovakia
hosmich@gmail.com, jiraskov@saske.sk
[2] Faculty of Electrical Engineering and Informatics, Technical University of Košice,
Boženy Němcovej 32, 042 00 Košice, Slovakia
mlynarcik1972@gmail.com

Abstract. A fooling set for a regular language is a special set of pairs of strings whose size provides a lower bound on the number of states in any nondeterministic finite automaton accepting this language. We show that, in spite of the fact that the difference between the size of the largest fooling set and the nondeterministic state complexity may be arbitrarily large, the fooling set lower bound methods work in many cases. We modify the method in the case when multiple initial states may save one state. We also state some useful properties that allow us to avoid describing particular fooling sets which may often be difficult and tedious.

1 Introduction

The nondeterministic state complexity of a regular language is the smallest number of states in any nondeterministic finite automaton (NFA) accepting this language. To get lower bounds on the nondeterministic state complexity, usually a fooling set technique is used. A fooling set is a special set of pairs of strings whose size provides a lower bound on the number of states in any NFA for a given language.

The lower bound method based on fooling sets, as a version of the crossing sequence argument, has been used for proving lower bounds on VLSI computations [4,14,15,19,22]. The fooling set method as a method providing lower bounds on communication complexity has been formulated by Aho, Ullman, and Yannakakis [1]. In the settings of formal languages, the method has been first described by Birget [2,3], and examined by Glaister and Shallit [6].

Research supported by VEGA grant 2/0084/15 and grant APVV-15-0091. This work was conducted during PhD study of Michal Hospodár at the Faculty of Mathematics, Physics and Informatics of the Comenius University in Bratislava, Slovakia.

H.-J. Böckenhauer et al. (Eds.): Hromkovič Festschrift, LNCS 11011, pp. 17–32, 2018.
https://doi.org/10.1007/978-3-319-98355-4_2

Although the gap between the size of a largest fooling set and the nondeterministic state complexity of a regular language may be arbitrarily large [7], in many cases, the fooling set method provides lower bounds that are tight.

In [12,13,18,20], the nondeterministic state complexity of basic regular operations in the subclasses of convex languages has been investigated. The authors considered operations of union, intersection, concatenation, star, reversal, and complementation in the classes of prefix-, suffix-, factor-, and subword-free, -closed, and -convex languages, and right, left, two-sided, and all-sided ideals. For each operation and each class, except for complementation on factor- and subword-convex languages, tight upper bounds have been obtained, and to get lower bounds, a fooling set lower bound method has been used in each case.

Here we present the fooling set method as an effective tool for getting lower bounds on the nondeterministic state complexity. We state some sufficient properties on NFAs that guarantee the existence of a sufficiently large fooling set for the accepted language. As a result, we can avoid the description of a fooling set which may sometimes be rather difficult and tedious. Moreover, the size of a fooling set provides a lower bound on the size of NFAs even with multiple initial states. This means that, for example, in the case of union or reversal, where NFAs with multiple initial states may save one state in the resulting automaton, the method cannot yield matching bounds. We describe a modification of the method consisting in a possibility to divide a fooling set into two parts such that adding a pair with left component equal to the empty string results again in a fooling set. Since after reading the empty string, the NFA is in its unique initial state, this state must be different from all states given by the fooling set.

We start by restating the result from [7] that the gap between the size of a largest fooling set and the nondeterministic state complexity of a regular language may be arbitrarily large. Then we formulate two lemmas with sufficient conditions on NFAs that guarantee their minimality. We continue with a modification of the fooling set method providing lower bounds on the size of NFAs with a unique initial state. Finally, we give a sufficient condition for getting large lower bounds for the complementation operation.

2 Preliminaries

We assume that the reader is familiar with basic notions in formal languages and automata theory. For details and all the unexplained notions, the reader may refer to [11,23,24].

Let Σ be a finite non-empty alphabet of symbols. Then Σ^* denotes the set of strings over the alphabet Σ including the empty string ε. The length of a string w is denoted by $|w|$, and the number of occurrences of a symbol a in a string w by $|w|_a$. A language is any subset of Σ^*. For a finite set X, the cardinality of X is denoted by $|X|$, and its power set by 2^X.

A *nondeterministic finite automaton* (with a nondeterministic choice of initial states; cf. [24]) (NNFA) is a quintuple $A = (Q, \Sigma, \cdot, I, F)$, where Q is a finite non-empty set of states, Σ is a finite non-empty alphabet, $I \subseteq Q$ is the set of

initial states, $F \subseteq Q$ is the set of final (or accepting) states, and the function $\cdot : Q \times \Sigma \to 2^Q$ is the transition function which is naturally extended to the domain $2^Q \times \Sigma^*$. The *language accepted by* A is $L(A) = \{w \in \Sigma^* \mid I \cdot w \cap F \neq \emptyset\}$.

We say that (p, a, q) is a *transition* in A if $q \in p \cdot a$. If (p, a, q) is a transition in A, then we say that the state q has an *in-transition*, and the state p has an *out-transition*. We sometimes write $p \xrightarrow{w} q$ if $q \in p \cdot w$.

An NNFA A is a *trim* NNFA if each its state q is reachable and useful, that is, there are strings u and v in Σ^* such that $q \in I \cdot u$ and $q \cdot v \cap F \neq \emptyset$.

If $|I| = 1$, we say that A is a nondeterministic finite automaton (NFA). In an ε-NFA, we also allow transitions on the empty string. It is known that the ε-transitions can be removed without increasing the number of states in the resulting NFA (cf. [11, Theorem 2.2] and [24, Theorem 2.3]).

An NFA A is a *deterministic finite automaton* (DFA) if $|q \cdot a| = 1$ for each q in Q and each a in Σ. Next, A is a *partial deterministic finite automaton* if $|q \cdot a| \leq 1$ for each q in Q and each a in Σ.

Every NNFA $A = (Q, \Sigma, \cdot, I, F)$ can be converted to an equivalent DFA $\mathcal{D}(A) = (2^Q, \Sigma, \cdot, I, \{S \in 2^Q \mid S \cap F \neq \emptyset\})$. We call the DFA $\mathcal{D}(A)$ the *subset automaton* of the NNFA A. The subset automaton might not be minimal since some of its states may be unreachable or equivalent to other states.

The *nondeterministic state complexity of a regular language* L, $\mathrm{nsc}(L)$, is the smallest number of states in any NFA accepting L. The *nondeterministic state complexity of a regular operation* is the number of states that are sufficient and necessary in the worst case for an NFA to accept the language resulting from the operation, considered as a function of the number of states of NFAs for the given operands. Formally, the nondeterministic state complexity of a binary regular operation $\circ$ is a function from $\mathbb{N}^2$ to $\mathbb{N}$ defined as

$$(m, n) \mapsto \max\{\mathrm{nsc}(K \circ L) \mid \mathrm{nsc}(K) \leq m \text{ and } \mathrm{nsc}(L) \leq n\}.$$

For a language L over an alphabet Σ, the *complement* of L is the language $L^c = \Sigma^* \setminus L$. The *intersection* of languages K and L is the language $K \cap L = \{w \mid w \in K \text{ and } w \in L\}$. The *union* of languages K and L is the language $K \cup L = \{w \mid w \in K \text{ or } w \in L\}$. The *concatenation* of languages K and L is the language $KL = \{uv \mid u \in K \text{ and } v \in L\}$. The *(Kleene) star* of a language L is the language $L^* = \bigcup_{i \geq 0} L^i$ where $L^0 = \{\varepsilon\}$ and $L^i = LL^{i-1}$ if $i \geq 1$.

The reversal w^R of a string w is defined as $\varepsilon^R = \varepsilon$ and $(wa)^R = aw^R$ for each symbol a and string w. The reversal of a language L is $L^R = \{w^R \mid w \in L\}$. If a language L is accepted by an NNFA $A = (Q, \Sigma, \cdot, I, F)$, then the language L^R is accepted by the NNFA A^R obtained from A by reversing all the transitions, and by swapping the roles of the initial and final states. Formally, we have $A^R = (Q, \Sigma, \cdot^R, F, I)$ where $q \cdot^R a = \{p \in Q \mid q \in p \cdot a\}$.

Let $A = (Q, \Sigma, \cdot, I, F)$ be an NNFA and $S, T \subseteq Q$. We say that S is *reachable* in A if there is a string w in Σ^* such that $S = I \cdot w$. Next, we say that T is *co-reachable* in A if T is reachable in A^R. Notice that if T is co-reachable in A, then there is a string w in Σ^* such that w is accepted by A from each state in T and rejected from each state in T^c.

If $u, v, w, x \in \Sigma^*$ and $w = uxv$, then u is a *prefix* of w, x is a *factor* of w, and v is a *suffix* of w. If $w = u_0v_1u_1 \cdots v_nu_n$, where $u_i, v_i \in \Sigma^*$, then $v_1v_2 \cdots v_n$ is a *subword* of w. A prefix v (suffix, factor, subword) of w is *proper* if $v \neq w$.

A language L is *prefix-free* if $w \in L$ implies that no proper prefix of w is in L; it is *prefix-closed* if $w \in L$ implies that each prefix of w is in L; and it is *prefix-convex* if $u, w \in L$ and u is a prefix of w imply that each string v such that u is a prefix of v and v is a prefix of w is in L. Suffix-, factor-, and subword-free, -closed, and -convex languages are defined analogously.

A language L is a right (respectively, left, two-sided, all-sided) *ideal* if $L = L\Sigma^*$ (respectively, $L = \Sigma^*L, L = \Sigma^*L\Sigma^*, L = L \sqcup \Sigma^*$), where $\sqcup$ denotes the shuffle operation [5]. Notice that the classes of free, closed, and ideal languages are subclasses of convex languages.

3 Fooling Set Lower Bound Method

To get lower bounds on the number of states in an NNFA accepting a regular language, the fooling set technique has been successfully used in the literature. We start with the definition of a fooling set, and with a lemma showing that the size of a fooling set for a regular language provides a lower bound on the nondeterministic state complexity of this language.

Definition 1. *A set of pairs of strings* $\{(x_i, y_i) \mid i = 1, 2, \ldots, n\}$ *is called a* fooling set for a language L *if for each* i, j *in* $\{1, 2, \ldots, n\}$,

(1) $x_iy_i \in L$, *and*
(2) if $i \neq j$, *then* $x_iy_j \notin L$ *or* $x_jy_i \notin L$.

Example 1. Let $L = \{a^n\}$ where $n \geq 0$. Consider the set of pairs of strings $\mathcal{F} = \{(a^i, a^{n-i}) \mid i = 0, 1, \ldots, n\}$. The string a^n is in L, while for each k with $k \neq n$, the string a^k is not in L. It follows that $\mathcal{F}$ is a fooling set for L. □

Lemma 1 (Birget [2, Lemma 1]**).** *Let* $\mathcal{F}$ *be a fooling set for a regular language* L. *Then every* NNFA *for* L *has at least* $|\mathcal{F}|$ *states.*

Proof. Let $\mathcal{F} = \{(x_i, y_i) \mid i = 1, 2, \ldots, n\}$ be a fooling set for L and A be an NNFA for L. For each i, fix an accepting computation of A on x_iy_i. Let p_i be the state on this computation reached after reading x_i, that is, we have

$$q_i \xrightarrow{x_i} p_i \xrightarrow{y_i} f_i$$

for an initial state q_i and a final state f_i. We now show that the states $p_1, p_2, \ldots, p_n$ are pairwise distinct. Let $i \neq j$ and suppose for a contradiction that $p_i = p_j$. However, in such a case, the NNFA A has the following accepting computation on both x_iy_j and x_jy_i:

$$q_i \xrightarrow{x_i} p_i = p_j \xrightarrow{y_j} f_j \text{ and } q_j \xrightarrow{x_j} p_j = p_i \xrightarrow{y_i} f_i.$$

This is a contradiction with condition (2) of Definition 1. □

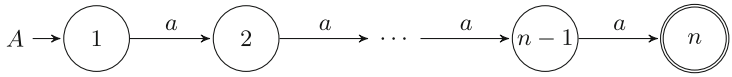

Fig. 1. A partial DFA in which each set $\{i\}$ is reachable and co-reachable.

Our first aim is to show that a gap between the maximal size of the fooling set and the nondeterministic state complexity of a language can be arbitrarily large. To this end, we introduce the notion of a *fooling set for an automaton.*

Definition 2. *A set of pairs of sets of states* $\mathcal{S} = \{(X_i, Y_i) \mid i = 1, 2, \ldots, n\}$ *is called a* fooling set for an NNFA A *if, for each* i, j *in* $\{1, 2, \ldots, n\}$,

(1) X_i *is reachable and* Y_i *is co-reachable in* A,
(2) $X_i \cap Y_i \neq \emptyset$, *and*
(3) if $i \neq j$, *then* $X_i \cap Y_j = \emptyset$ *or* $X_j \cap Y_i = \emptyset$.

Example 2. Let A be the partial DFA shown in Fig. 1. The set of pairs of sets

$$\{(\{i\}, \{i\}) \mid i = 1, 2, \ldots, n\}$$

is a fooling set for A since each $\{i\}$ is reachable and co-reachable in A, and conditions (2) and (3) of Definition 2 are satisfied as well. □

Proposition 1. *Let* A *be an* NNFA. *Then a fooling set of size* n *for the language* $L(A)$ *exists if and only if a fooling set of size* n *for the automaton* A *exists.*

Proof. Let $A = (Q, \Sigma, \cdot, I, F)$ be an NNFA. Let $\mathcal{F} = \{(x_i, y_i) \mid i = 1, 2, \ldots, n\}$ be a fooling set for $L(A)$. Set $X_i = I \cdot x_i$ and $Y_i = F \cdot^R y_i^R$ for $i = 1, 2, \ldots, n$. Then $\mathcal{S} = \{(X_i, Y_i) \mid i = 1, 2, \ldots, n\}$ is the desired fooling set for A.

Conversely, since X_i is reachable, there is a string x_i such that $X_i = I \cdot x_i$. Since Y_i is co-reachable, there is a string y_i such that $Y_i = F \cdot^R y_i$. Then the set $\{(x_i, y_i^R) \mid i = 1, 2, \ldots, n\}$ is a fooling set for $L(A)$. □

Theorem 1 (cf. Gruber and Holzer [7, Theorem 10]**).** *Let* $n \geq 4$. *There exists a language* L *such that every fooling set for* L *is of size at most 3, while every* NFA *for* L *has at least* $\log_2 n$ *states.*

Proof. Let L be the unary language accepted by the DFA A with the state set $Q = \{1, 2, \ldots, n\}$ shown in Fig. 2. The DFA A has n states and it is a minimal DFA for L. It follows that every NFA for L must have at least $\log_2 n$ states.

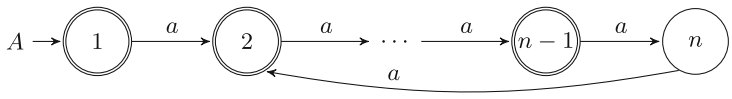

Fig. 2. The minimal DFA of a language with a fooling set of size at most 3 and with nondeterministic state complexity at least $\log_2 n$.

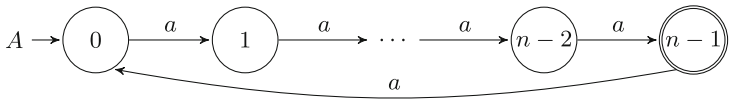

Fig. 3. A unary witness language for star meeting the upper bound $n + 1$.

Let us show that every fooling set for L is of size at most 3. Suppose for a contradiction that there is a fooling set for L of size 4. Then there is a fooling set $\mathcal{S} = \{(X_i, Y_i) \mid i = 1, 2, 3, 4\}$ for A. This means that for each i, the set X_i is reachable and the set Y_i is co-reachable in A with $X_i \cap Y_i \neq \emptyset$, and if $i \neq j$, then at least one of $X_i \cap Y_j$ and $X_j \cap Y_i$ is empty. Consider the bipartite graph $(\mathcal{R}, \mathcal{C}, E)$ where

- $\mathcal{R} = \{S \subseteq Q \mid S \text{ is reachable in } A\} = \{\{i\} \mid i = 1, 2, \ldots, n\}$,
- $\mathcal{C} = \{T \subseteq Q \mid T \text{ is co-reachable in } A\} = \{Q \setminus \{i\} \mid 2 \leq i \leq n\} \cup \{Q \setminus \{1, n\}\}$,
- $(S, T) \in E$ iff $S \cap T \neq \emptyset$.

Notice that each $\{i\}$ in $\mathcal{R}$, except for $\{n\}$, is connected to each set in $\mathcal{C}$, except for the set $Q \setminus \{i\}$. The set $\{n\}$ is connected to each set in $\mathcal{C}$, except for $Q \setminus \{n\}$ and $Q \setminus \{1, n\}$.

Now the set $\mathcal{S} = \{(X_i, Y_i) \mid i = 1, 2, 3, 4\}$ is a fooling set for A by assumption. Consider the subgraph G of $(\mathcal{R}, \mathcal{C}, E)$ induced by the set $\{X_i \mid i = 1, 2, 3, 4\} \cup \{Y_i \mid i = 1, 2, 3, 4\}$. If $X_i \neq \{n\}$, then X_i is not connected to at most one of Y_i, $i = 1, 2, 3, 4$. If $X_i = \{n\}$, then X_i is not connected to at most two of Y_i, $i = 1, 2, 3, 4$. This means that there are at least 11 edges in G. However, since $\mathcal{S}$ is a fooling set, for every two distinct pairs (X_i, Y_i) and (X_j, Y_j), at least one edge must be missing in G. In total, at least 6 edges must be missing in G – one for every two distinct pairs. However, this only gives 10 possible edges in G, a contradiction. □

Although the previous theorem shows that the gap between the size of a fooling set for a regular language and its nondeterministic state complexity may be arbitrarily large, our next aim is to show that in most cases, the fooling set technique provides lower bounds that are tight.

The next example illustrates how both types of fooling sets can be used to obtain the nondeterministic state complexity of the star operation. The upper bound on the nondeterministic state complexity of the star operation is $n + 1$ since we can construct an ε-NFA for the star of a language given by an NFA by adding a new initial and final state connected through ε-transitions to the original initial state, and by adding ε-transitions from every final state to the original initial state. The next example provides a unary witness language.

Example 3 (Star on regular languages; cf. [10, Theorem 9]*).* Let L be the unary language accepted by the n-state DFA A shown in Fig. 3, where $n \geq 2$. Let us show that every NFA for L^* has at least $n+1$ states. We prove this by describing (a) a fooling set for L^*, and (b) a fooling set for an NFA for L^*.

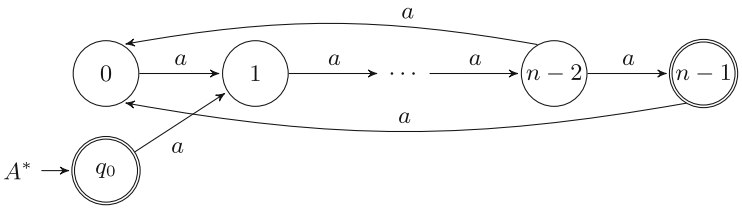

Fig. 4. The NFA A^* for the automaton A from Fig. 3.

(a) Consider the set of pairs of strings

$$\mathcal{F} = \{(\varepsilon, \varepsilon)\} \cup \{(a^i, a^{n-1-i}) \mid 1 \leq i \leq n-2\} \cup \{(a^{n-1}, a^n), (a^n, a^{n-1})\}.$$

We have $\{\varepsilon, a^{n-1}, a^{2n-1}\} \subseteq L^*$. However, if $1 \leq k \leq n-2$, then $a^k \notin L^*$ and $a^{n+k} \notin L^*$. Moreover, $a^n \notin L^*$ and $a^{2n} \notin L^*$. This means that $\mathcal{F}$ is a fooling set for L^*, so every NFA for L^* has at least $n+1$ states.

(b) Construct an NFA A^* for L^* from A by adding the transition $(n-2, a, 0)$, and by adding a new initial state q_0 going to the state 1 on a; see Fig. 4. Set

$$\begin{array}{llll} X_0 & = \{q_0\}, & Y_0 & = \{q_0, n-1\}, \\ X_i & = \{i\}, & Y_i & = \{i\} \qquad \text{for } i = 1, 2, \ldots, n-2, \\ X_{n-1} & = \{0, n-1\}, & Y_{n-1} & = \{n-2, n-1\}, \\ X_n & = \{0, 1\}, & Y_n & = \{0, q_0\}. \end{array}$$

Then for each $i = 0, 1, \ldots, n$, the set X_i is reachable and the set Y_i is co-reachable in A^*. Next, if $(i, j) \in \{(0, n), (n-2, n-1), (n-1, n)\}$, then $X_j \cap Y_i = \emptyset$, otherwise, $X_i \cap Y_j = \emptyset$ if $i < j$. It follows that the set $\{(X_i, Y_i) \mid i = 0, 1, \ldots, n\}$ is a fooling set for A^*, so every NNFA for L^* has at least $n+1$ states. □

4 Simplifications of the Fooling Set Method

In this section, we state some sufficient conditions on an NNFA that guarantee its minimality. Having such an NNFA, there is no need to describe a fooling set for the accepted language since we know that every equivalent NNFA has at least as many states as the given NNFA.

Lemma 2. *Let $A = (Q, \Sigma, \cdot, I, F)$ be an* NNFA. *Suppose that, for each state q in Q, the one-element set $\{q\}$ be reachable as well as co-reachable in A. Then every* NNFA *for $L(A)$ has at least $|Q|$ states.*

Proof. Since $\{q\}$ is reachable in A, there is a string x_q such that $I \cdot x_q = \{q\}$. Since $\{q\}$ is co-reachable in A, there is a string y_q accepted by A from and only from the state q. Then $\{(x_q, y_q) \mid q \in Q\}$ is a fooling set for the language $L(A)$. By Lemma 1, every NNFA for $L(A)$ has at least $|Q|$ states. □

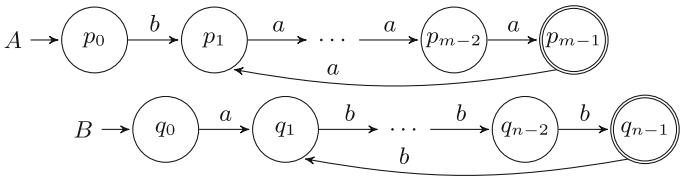

Fig. 5. Binary suffix-free witnesses for union meeting the upper bound $m + n - 1$.

Notice that if A is a trim partial DFA, then for each state q of A, the singleton set $\{q\}$ is reachable. If moreover A^R is a partial DFA, then $\{q\}$ is co-reachable in A. So we get the following result.

Lemma 3. *Let A be an n-state trim* NFA. *If both A and A^R are partial* DFAs, *then every* NNFA *for $L(A)$ has at least n states.* □

Let us show how Lemma 2 can be used to get the nondeterministic state complexity of the union operation on suffix-free languages. Recall that if two NFAs A and B accept suffix-free languages, then we may assume that their initial states do not have any in-transitions [8,21] This means that we can merge the initial states to get an NFA for $L(A) \cup L(B)$. This gives an upper bound of $m + n - 1$. In the next example, we use Lemma 2 to prove the tightness of this upper bound.

Example 4 (Union on suffix-free languages; cf. [13, Theorem 9]*).* Consider the NFAs A and B shown in Fig. 5; notice that the languages $L(A)$ and $L(B)$ are suffix-free. Construct an NFA for $L(A) \cup L(B)$ by merging the initial states of A and B; see Fig. 6. For each state q of the resulting NFA, the set $\{q\}$ is reachable, as well as co-reachable. By Lemma 2, every NNFA for $L(A) \cup L(B)$ has at least $m + n - 1$ states. □

Now, we use Lemma 3 to get the nondeterministic complexity of intersection on regular languages. The upper bound is mn since the product automaton $A \times B = (Q_A \times Q_B, \Sigma, \cdot, (s_A, s_B), F_A \times F_B)$, where $(p, q) \cdot a = (p \cdot_A a) \times (q \cdot_B a)$, recognizes $L(A) \cap L(B)$. The next example provides binary witness languages.

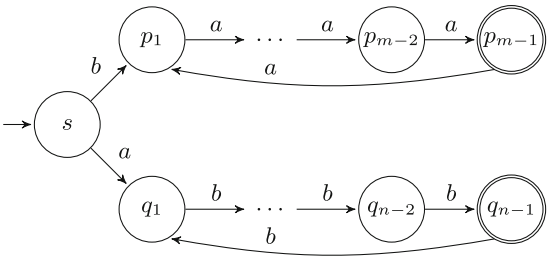

Fig. 6. The NFA for the union of languages from Fig. 5.

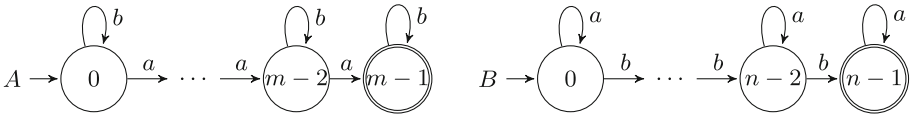

Fig. 7. Binary witness languages for intersection meeting the upper bound mn.

Example 5 (Intersection on regular languages; cf. [10, Theorem 3]*).* Consider the partial DFAs A and B shown in Fig. 7. The product automaton $A \times B$ for $L(A) \cap L(B)$ is a trim partial DFA, and its reverse is a partial DFA as well; see Fig. 8 for $m = 3$ and $n = 4$. By Lemma 3, every NNFA for $L(A) \cap L(B)$ has at least mn states. □

5 Modification of the Fooling Set Method

The fooling set method provides a lower bound on the number of states in any NNFA, that is, in any nondeterministic finite automaton with, possibly, multiple initial states. However, sometimes the NFA with a unique initial state must have one additional state. This is true for the case of union, where for every pair of languages K and L accepted by an m-state and n-state NFA, respectively, there exists an NNFA of size $m + n$ accepting $K \cup L$, hence no fooling set for $K \cup L$ can be of size more than $m + n$. Similarly, no fooling set for L^R can be of size more than n.

The idea for getting lower bounds of $m + n + 1$ for union and of $n + 1$ for reversal, is to divide a fooling set into two parts $\mathcal{A}$ and $\mathcal{B}$ and to find pairs (ε, u) and (ε, v) such that $\mathcal{A} \cup \{(\varepsilon, u)\}$ and $\mathcal{B} \cup \{(\varepsilon, v)\}$ are fooling sets. This implies that a unique initial state, reached after reading the empty string, must be different from all states given by fooling set $\mathcal{A} \cup \mathcal{B}$.

Lemma 4 (Jirásková and Masopust [17, Lemma 4]**).** *Let $\mathcal{A}$ and $\mathcal{B}$ be disjoint sets of pairs of strings and let u and v be two strings such that $\mathcal{A} \cup \mathcal{B}$, $\mathcal{A} \cup \{(\varepsilon, u)\}$, and $\mathcal{B} \cup \{(\varepsilon, v)\}$ are fooling sets for a language L. Then every* NFA *for L has at least $|\mathcal{A}| + |\mathcal{B}| + 1$ states.*

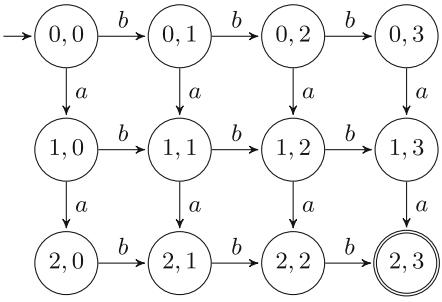

Fig. 8. The product automaton for the intersection of languages from Fig. 7, for $m = 3$ and $n = 4$.

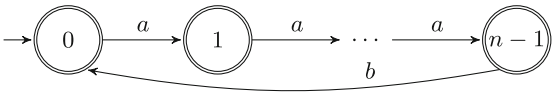

Fig. 9. A binary witness language for reversal meeting the upper bound $n + 1$.

Proof. Let A be an NFA for L with the initial state s. Let

$$\mathcal{A} = \{(x_i, y_i) \mid i = 1, 2, \ldots, m\}, \text{ and}$$
$$\mathcal{B} = \{(x_{m+j}, y_{m+j}) \mid j = 1, 2, \ldots, n\}.$$

Since each string $x_k y_k$ is in L, we can fix an accepting computation of A on $x_k y_k$, and let p_k be the state on this computation reached after reading x_k. Since $\mathcal{A} \cup \mathcal{B}$ is a fooling set for L, the states $p_1, p_2, \ldots, p_{m+n}$ are pairwise distinct, as shown in the proof of Lemma 1. Since $\mathcal{A} \cup \{(\varepsilon, u)\}$ is a fooling set, the initial state s is distinct from all the states $p_1, p_2, \ldots, p_m$. Since $\mathcal{B} \cup \{(\varepsilon, v)\}$ is a fooling set, the initial state s is also distinct from all the states $p_{m+1}, p_{m+2}, \ldots, p_{m+n}$. Thus the NFA A has at least $m + n + 1$ states. □

It is shown in [16, Theorem 2] that there is a *binary* regular language L accepted by an n-state NFA such that every NFA for L^R has at least $n + 1$ states. An NFA for the language is shown in Fig. 9, and the proof in [16] is by a counting argument. In the next example we use Lemma 4 to get the lower bound.

Example 6 (Reversal on regular languages; cf. [16, Theorem 2]*).* Let L be the binary language accepted by the partial DFA shown in Fig. 9. Set

$$\mathcal{A} = \{(ba^i, a^{n-1-i}) \mid i = 0, 1, \ldots, n-2\},$$
$$\mathcal{B} = \{(ba^{n-1}, \varepsilon)\},$$
$$u = \varepsilon,$$
$$v = a.$$

Notice that we have $\{ba^{n-1}, \varepsilon, a\} \subseteq L^R$. On the other hand, if $k \neq n - 1$, then the string ba^k is not in L^R. It follows that $\mathcal{A} \cup \mathcal{B}$, $\mathcal{A} \cup \{(\varepsilon, u)\}$, and $\mathcal{B} \cup \{(\varepsilon, v)\}$ are fooling sets for the language L^R. By Lemma 4, every NFA for L^R has at least $n + 1$ states. Since $n + 1$ is also an upper bound on the nondeterministic state complexity of the reversal operation, we get $\mathrm{nsc}(L^R) = n + 1$. □

The following two examples use Lemma 4 to get the nondeterministic state complexity of the union operation on regular and prefix-free languages.

Example 7 (Union on regular languages; cf. [10, Theorem 1]*).* Let $K = \{a^m\}^*$ and $L = \{b^n\}^*$ where $m, n \geq 1$. The languages K and L are accepted by the m-state and n-state NFA A and B, respectively, shown in Fig. 10. It is shown in [10, Theorem 1] that every NFA for the language $K \cup L$ has at least $m + n + 1$

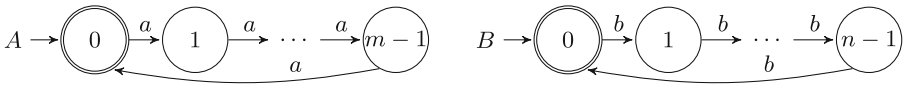

Fig. 10. Binary witness languages for union meeting the upper bound $m + n + 1$.

states, and the proof is almost one page long. Let us again use Lemma 4. To this end, let

$$\begin{aligned}
\mathcal{A} &= \{(a^i, a^{m-i}) \mid i = 1, 2, \ldots, m-1\} \cup \{(a^m, a^m)\},\\
\mathcal{B} &= \{(b^j, b^{n-j}) \mid j = 1, 2, \ldots, n-1\} \cup \{(b^n, b^n)\},\\
u &= b^n,\\
v &= a^m.
\end{aligned}$$

We have $\{a^m, a^{2m}, b^n, b^{2n}\} \subseteq K \cup L$. On the other hand, if $k \not\equiv 0 \bmod m$, then $a^k \notin K \cup L$, and similarly, if $\ell \not\equiv 0 \bmod n$, then $b^\ell \notin K \cup L$. Moreover, no string in $K \cup L$ contains both a and b. It follows that $\mathcal{A} \cup \mathcal{B}$, $\mathcal{A} \cup \{(\varepsilon, u)\}$, and $\mathcal{B} \cup \{(\varepsilon, v)\}$ are fooling sets for the language $K \cup L$. By Lemma 4, every NFA for $K \cup L$ has at least $m + n + 1$ states. □

Example 8 (Union on prefix-free languages; cf. [13, Theorem 9]*).* A minimal NFA for a prefix-free language has exactly one final state with no out-transitions. We can construct an NNFA for the union of prefix-free languages given by minimal NFAs A and B (with disjoint states sets) just by merging their final states. This gives $m + n - 1$ states in the resulting NNFA. Therefore $m + n$ is an upper bound on the nondeterministic complexity of the union of prefix-free languages.

In [9] it is claimed that the upper bound $m + n$ is met by the union of the prefix-free languages $K = (a^{m-1})^* b$ and $L = (c^{n-1})^* d$, and a set P of pairs of strings of size $m+n$ is described in [9, Proof of Theorem 3.2]. The authors claim that the set P is a fooling set for $K \cup L$. However, as shown above, the language $K \cup L$ is accepted by an NNFA of $m + n - 1$ states. Therefore P cannot be a fooling set for $K \cup L$; indeed, the pairs $(\varepsilon, a^{m-1}b)$ and (a^{m-1}, b) do not satisfy the second condition in Definition 1.

Here we prove the tightness of the upper bound $m+n$ for the union of prefix-free languages using a binary alphabet and Lemma 4. Let $m, n \geq 3$ and consider binary languages K and L accepted by the m-state and n-state NFA A and B, respectively, shown in Fig. 11. Notice that K is prefix-free since every string in K ends with b while every proper prefix of every string in K is in a^*. Now, using Lemma 4, we show that every NFA for $K \cup L$ has at least $m + n$ states.

Fig. 11. Binary prefix-free witnesses for union meeting the upper bound $m + n$.

To this end, let

$$\begin{aligned}
\mathcal{A} &= \{(a^{m-1+i}, a^{m-2-i}b) \mid 0 \le i \le m-2\} \cup \{(a^{m-2}b, \varepsilon)\},\\
\mathcal{B} &= \{(b^{n-1+j}, b^{n-2-j}a) \mid 0 \le j \le n-2\},\\
u &= b^{n-2}a,\\
v &= a^{m-2}b.
\end{aligned}$$

We have $\{a^{2m-3}b, a^{m-2}b, b^{2n-3}a, b^{n-2}a\} \subseteq K \cup L$. Next, every string in the language $K \cup L$ starting with a has $(m-2)$ mod $(m-1)$ consecutive a's followed by b, and every string starting with b has $(n-2)$ mod $(n-1)$ consecutive b's followed by a. This means that the sets $\mathcal{A} \cup \mathcal{B}$, $\mathcal{A} \cup \{(\varepsilon, u)\}$ and $\mathcal{B} \cup \{(\varepsilon, v)\}$ are fooling sets for $K \cup L$. By Lemma 4, every NFA for $K \cup L$ has at least $m + n$ states. □

Finally, we use Lemma 4 to show that the upper bound $m + n + 1$ for the union of regular languages can be met by binary subword-closed languages.

Example 9 (Union on subword-closed languages; [12, Theorem 4]*).* Let $m, n \ge 1$ and $K = \{w \in \{a, b\}^* \mid |w|_a \le m-1\}$ and $L = \{w \in \{a, b\}^* \mid |w|_b \le n-1\}$ be the binary subword-closed languages accepted by the m-state and n-state partial DFA A and B, respectively, shown in Fig. 12. Let

$$\begin{aligned}
\mathcal{A} &= \{(b^n a^i, a^{m-1-i}) \mid 0 \le i \le m-1\},\\
\mathcal{B} &= \{(b^{n-1-j}, b^j a^m) \mid 0 \le j \le n-1\},\\
u &= a^m b^{n-1},\\
v &= a^{m-1} b^n.
\end{aligned}$$

Each string w with $|w|_a = m-1$ or $|w|_b = n-1$ is in $K \cup L$, while no string with $|w|_a \ge m$ and $|w|_b \ge n$ is in $K \cup L$. It follows that $\mathcal{A} \cup \mathcal{B}$, $\mathcal{A} \cup \{(\varepsilon, u)\}$, and $\mathcal{B} \cup \{(\varepsilon, v)\}$ are fooling sets for $K \cup L$. By Lemma 4, every NFA for $K \cup L$ has at least $m + n + 1$ states. □

6 Fooling Sets for Complementation

The next very useful observation allows us to significantly simplify the proofs of lower bounds on the nondeterministic state complexity of complementation.

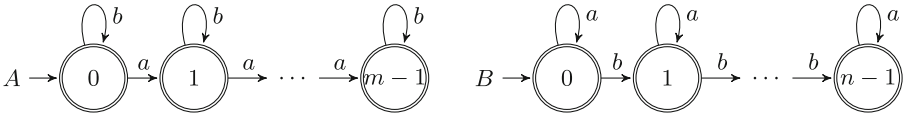

Fig. 12. Binary subword-closed witnesses for union meeting the bound $m + n + 1$.

Lemma 5. *Let A be an n-state* NNFA *in which each subset of the state set is reachable as well as co-reachable. Then every* NNFA *for the complement of the language $L(A)$ has at least 2^n states.*

Proof. Let $A = (Q, \Sigma, \cdot, I, F)$ be an NNFA. Let $S \subseteq Q$. Since S is reachable, there exists a string x_S in Σ^* such that $S = I \cdot x_S$. Since the set S^c is co-reachable, there is a string y_S which is accepted by A from each state in S^c, but rejected from each state in S. It follows that the set $\{(x_S, y_S) \mid S \subseteq Q\}$ is a fooling set for $(L(A))^c$. Hence every NNFA for $(L(A))^c$ has at least 2^n states. □

Recall that to get an NFA (even DFA) for the complement of a language represented by an NFA A, we first apply the subset construction to the NFA A, and in the resulting DFA, we interchange the accepting and rejecting states. This gives an upper bound of 2^n on the nondeterministic state complexity of complementation. In the next example we use Lemma 5 to show that the complement of the binary language from [16, Proof of Theorem 5] meets this upper bound. Notice that a rather complicated fooling set is described in [16], and the proof is almost three pages long. Moreover, the NFA for this binary witness and its reverse are isomorphic, so we only need to show reachability of all subsets.

Example 10 (Complementation on regular languages [16, Theorem 5]*).* Consider the n-state NFA A shown in Fig. 13, where $i \cdot a = \{i+1\}$ and $i \cdot b = \{0, i+1\}$ if $0 \leq i \leq n-2$, $(n-1) \cdot b = \{1, 2, \ldots, n-1\}$, and all the remaining transitions go to the empty set. First, notice that the automata A and A^R are isomorphic. Hence to prove that every NFA for $(L(A))^c$ has at least 2^n states, we only need to show that every subset of $\{0, 1, \ldots, n-1\}$ is reachable in A. Since every subset S with $0 \notin S$ can be reached by $a^{\min S}$ from the set $\{s - \min S \mid s \in S\}$ which contains state 0, we only need to show the reachability of subsets containing 0. The proof is by induction on the size of subsets. The set $\{0\}$ is the initial subset, and every

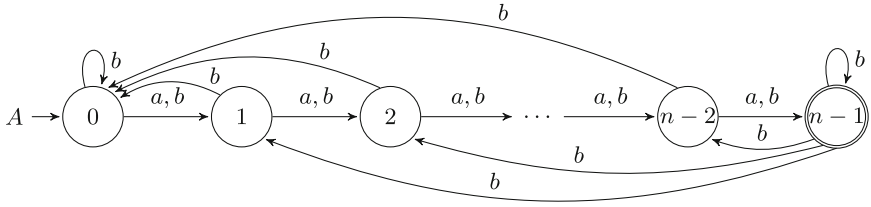

Fig. 13. A binary witness for complementation meeting the upper bound 2^n.

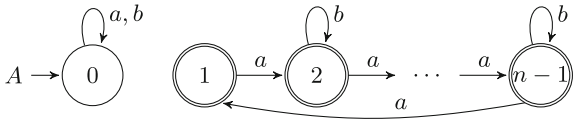

Fig. 14. Transitions on a and b in a suffix-convex witness for complementation.

subset $\{0, i_2, \ldots, i_k\}$ of size k, where $1 \leq i_2 < \cdots < i_k \leq n-1$, is reachable from the set $\{0, i_3 - i_2, \ldots, i_k - i_2\}$ of size $k-1$ by the string ab^{i_2-1}. □

Finally, we prove that the upper bound 2^n can be met by the complement of a suffix-convex language. Let us emphasize that such a language must be so-called *proper suffix-convex*, that is, it can be neither suffix-free, nor suffix-closed, nor left ideal since it is proved in [18, Lemma 4, Theorem 2], [12, Theorem 10], and [20, Theorem 26], respectively, that the nondeterministic complexity of complementation is less than 2^n in these three subclasses of convex languages.

Example 11 (Complementation on suffix-convex languages; [13, Theorem 13]*).* Let $n \geq 3$. Let L be the regular language accepted by the nondeterministic finite automaton $A = (\{0, 1, \ldots, n-1\}, \{a, b, c, d, e\}, 0, \cdot, \{1, 2, \ldots, n-1\})$ where the transitions on a and b are shown in Fig. 14, the transitions on c, d, e are as follows:

$$\begin{aligned} 0 \cdot c &= \{0, 1, \ldots, n-1\}, \\ 0 \cdot d &= \{1, 2, \ldots, n-1\}, \\ q \cdot e &= \{n-1\}, \text{ for each state } q \text{ of } A, \end{aligned}$$

and all the remaining transitions go to the empty set. In the NFA A^R, the final state 0 loops on a, b, c and goes to the empty set on d and e. Next, every other state of A^R goes to 0 on d, and the state $n-1$ goes to $\{0, 1, \ldots, n-1\}$ on e. Thus in the subset automaton of A^R, each final subset, that is, a subset containing the state 0, goes either to a final subset containing 0 or to the empty set on each input symbol. It follows that the language L^R is prefix-convex, so L is suffix-convex. Now we show that each subset of the state set of A is reachable and co-reachable in A. Notice that $\{0\} \cdot a = \{0\}$, $\{0\} \cdot b = \{0\}$, $0 \cdot c = \{0, 1, \ldots, n-1\}$, and $0 \cdot d = \{1, 2, \ldots, n-1\}$.

Moreover, we can shift each subset of $\{1, 2, \ldots, n-1\}$ cyclically by one using the symbol a, that is, we have $S \cdot a = \{(s+1) \bmod n \mid s \in S\}$. Next, we can eliminate the state 1 from each subset containing 1 by reading the symbol b. It follows that each subset is reachable. To prove co-reachability, notice that the initial subset of A^R is $\{1, 2, \ldots, n-1\}$ and it goes to $\{0, 1, \ldots, n-1\}$ on e. We again use symbol a to shift the subsets of $\{1, 2, \ldots, n-1\}$ and symbol b to eliminate the state 1. It follows that every subset is co-reachable. By Lemma 5, every NNFA for L^c has at least 2^n states. □

7 Conclusions

The fooling set method provides lower bounds on the number of states in nondeterministic finite automata that are tight in many cases despite the fact that the gap between the size of a fooling set and the nondeterministic state complexity may be arbitrarily large. We illustrated this on a number of examples.

We also provided sufficient conditions on nondeterministic finite automata that guarantee the existence of appropriate fooling sets. This allowed us to avoid the tedious description of such fooling sets.

Since fooling sets provide lower bounds on the number of states in nondeterministic finite automata with multiple initial states, in the case of union or reversal, where such automata may save one state, the fooling set method cannot be used. However, if a fooling set can be divided into two parts such that adding a pair with its left component equal to the empty string results in another fooling set, we get tight lower bounds also for automata with a unique initial state.

Finally, we provided a very useful observation from [13, Proof of Theorem 13] claiming that if all subsets of the state set of a nondeterministic finite automaton are reachable and co-reachable, then the accepted language is a witness for the complementation operation

References

1. Aho, A.V., Ullman, J.D., Yannakakis, M.: On notions of information transfer in VLSI circuits. In: Proceedings of the Fifteenth Annual ACM Symposium on Theory of Computing, STOC 1983, pp. 133–139. ACM, New York (1983)
2. Birget, J.: Intersection and union of regular languages and state complexity. Inf. Process. Lett. **43**(4), 185–190 (1992)
3. Birget, J.: Partial orders on words, minimal elements of regular languages and state complexity. Theor. Comput. Sci. **119**(2), 267–291 (1993)
4. Brent, R.P., Kung, H.T.: The chip complexity of binary arithmetic. In: Proceedings of the Twelfth Annual ACM Symposium on Theory of Computing, STOC 1980, pp. 190–200. ACM, New York (1980)
5. Brzozowski, J., Jirásková, G., Liu, B., Rajasekaran, A., Szykuła, M.: On the state complexity of the shuffle of regular languages. In: Câmpeanu, C., Manea, F., Shallit, J. (eds.) DCFS 2016. LNCS, vol. 9777, pp. 73–86. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-41114-9_6
6. Glaister, I., Shallit, J.: A lower bound technique for the size of nondeterministic finite automata. Inf. Process. Lett. **59**(2), 75–77 (1996)
7. Gruber, H., Holzer, M.: Finding lower bounds for nondeterministic state complexity is hard. In: Ibarra, O.H., Dang, Z. (eds.) DLT 2006. LNCS, vol. 4036, pp. 363–374. Springer, Heidelberg (2006). https://doi.org/10.1007/11779148_33
8. Han, Y., Salomaa, K.: Nondeterministic state complexity for suffix-free regular languages. In: McQuillan, I., Pighizzini, G. (eds.) Proceedings Twelfth Annual Workshop on Descriptional Complexity of Formal Systems, DCFS 2010. EPTCS, vol. 31, pp. 189–196 (2010)
9. Han, Y., Salomaa, K., Wood, D.: Nondeterministic state complexity of basic operations for prefix-free regular languages. Fundam. Inform. **90**(1–2), 93–106 (2009)

10. Holzer, M., Kutrib, M.: Nondeterministic descriptional complexity of regular languages. Int. J. Found. Comput. Sci. **14**(6), 1087–1102 (2003)
11. Hopcroft, J.E., Ullman, J.D.: Introduction to Automata Theory, Languages and Computation. Addison-Wesley, Boston (1979)
12. Hospodár, M., Jirásková, G., Mlynárčik, P.: Nondeterministic complexity of operations on closed and ideal languages. In: Han, Y.-S., Salomaa, K. (eds.) CIAA 2016. LNCS, vol. 9705, pp. 125–137. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-40946-7_11
13. Hospodár, M., Jirásková, G., Mlynárčik, P.: Nondeterministic complexity of operations on free and convex languages. In: Carayol, A., Nicaud, C. (eds.) CIAA 2017. LNCS, vol. 10329, pp. 138–150. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-60134-2_12
14. Hromkovič, J.: Some complexity aspects of VLSI computations, part 1. Comput. Artif. Intell. **7**(3), 229–252 (1988)
15. Jirásková, G.: Comparison of two VLSI models. Comput. Artif. Intell. **10**, 121–232 (1991)
16. Jirásková, G.: State complexity of some operations on binary regular languages. Theor. Comput. Sci. **330**(2), 287–298 (2005)
17. Jirásková, G., Masopust, T.: Complexity in union-free regular languages. Int. J. Found. Comput. Sci. **22**(7), 1639–1653 (2011)
18. Jirásková, G., Mlynárčik, P.: Complement on prefix-free, suffix-free, and non-returning NFA languages. In: Jürgensen, H., Karhumäki, J., Okhotin, A. (eds.) DCFS 2014. LNCS, vol. 8614, pp. 222–233. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-09704-6_20
19. Lipton, R.J., Sedgewick, R.: Lower bounds for VLSI. In: Proceedings of the Thirteenth Annual ACM Symposium on Theory of Computing, STOC 1981, pp. 300–307. ACM, New York, NY, USA (1981)
20. Mlynárčik, P.: Complement on free and ideal languages. In: Shallit, J., Okhotin, A. (eds.) DCFS 2015. LNCS, vol. 9118, pp. 185–196. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-19225-3_16
21. Mlynárčik, P.: Nondeterministic state complexity in subregular classes. Dissertation thesis. FMFI UK, Bratislava (2017). http://im.saske.sk/~jiraskov/students/phd_thesis_mlynarcik.pdf
22. Savage, J.E.: Planar circuit complexity and the performance of VLSI algorithms +. In: Kung, H.T., Sproull, B., Steele, G. (eds.) VLSI Systems and Computations, pp. 61–68. Springer, Heidelberg (1981). https://doi.org/10.1007/978-3-642-68402-9_8
23. Sipser, M.: Introduction to the theory of computation. Cengage Learning (2012)
24. Yu, S.: Regular languages. In: Rozenberg, G., Salomaa, A. (eds.) Handbook of Formal Languages, vol. I, 1st edn. Springer, Heidelberg (1997). https://doi.org/10.1007/978-3-642-59136-5_2