

# Operations on Unambiguous Finite Automata

Jozef Jirásek Jr.<sup>1,2</sup>, Galina Jirásková<sup>1(✉)</sup>, and Juraj Šebej<sup>2</sup>

<sup>1</sup> Mathematical Institute, Slovak Academy of Sciences, Grešákova 6,  
040 01 Košice, Slovakia

jirasekjozef@gmail.com, jiraskov@saske.sk

<sup>2</sup> Faculty of Science, Institute of Computer Science, P.J. Šafárik University,  
Jesenná 5, 040 01 Košice, Slovakia

juraj.sebej@gmail.com

**Abstract.** A nondeterministic finite automaton is unambiguous if it has at most one accepting computation on every input string. We investigate the complexity of basic regular operations on languages represented by unambiguous finite automata. We get tight upper bounds for intersection ( $mn$ ), left and right quotients ( $2^n - 1$ ), positive closure ( $\frac{3}{4} \cdot 2^n - 1$ ), star ( $\frac{3}{4} \cdot 2^n$ ), shuffle ( $2^{mn} - 1$ ), and concatenation ( $\frac{3}{4} \cdot 2^{m+n} - 1$ ). To prove tightness, we use a binary alphabet for intersection and left and right quotients, a ternary alphabet for star and positive closure, a five-letter alphabet for shuffle, and a seven-letter alphabet for concatenation. We also get some partial results for union and complementation.

## 1 Introduction

A nondeterministic machine is unambiguous if it has at most one accepting computation on every input string. Ambiguity was studied intensively mainly in connection with context-free languages and it is well known that the classes of ambiguous, unambiguous, and deterministic context-free languages are all different. Ambiguity in finite automata was first considered by Schmidt [21] in his unpublished thesis, where he obtained a lower bound  $2^{\Omega(\sqrt{n})}$  on the conversion of unambiguous finite automata into deterministic finite automata, as well as for the conversion of nondeterministic finite automata into unambiguous finite automata. He also developed an interesting lower bound method for the size of unambiguous automata based on the rank of certain matrices.

Stearns and Hunt [23] provided polynomial-time algorithms for the equivalence and containment problems for unambiguous finite automata (UFAs), and they extended them to ambiguity bounded by a fixed integer  $k$ . Chan and Ibarra [5] provided a polynomial space algorithm to decide, given a nondeterministic finite automaton (NFA), whether it is finitely ambiguous. They also showed that it is PSPACE-complete to decide, given an NFA  $M$  and an integer  $k$ , whether  $M$  is  $k$ -ambiguous.

---

G. Jirásková—Research supported by VEGA grant 2/0084/15 and grant APVV-15-0091.

J. Šebej—Research supported by VEGA grant 1/0142/15 and grant APVV-15-0091.

Ibarra and Ravikumar [12] defined the ambiguity function  $a_M(n): N \rightarrow N$  of an NFA  $M$  such that  $a_M(n)$  is the maximum number of distinct accepting computations of  $M$  on any string of length  $n$ , and they proved that the exponential ambiguity problem is decidable for NFAs. Weber and Seidl [24] showed that if an  $n$ -state NFA is finitely ambiguous, then it is at most  $5^{n/2}n^n$ -ambiguous. Allauzen et al. [1] considered  $\varepsilon$ -NFAs, and they showed that, given a trim  $\varepsilon$ -cycle-free NFA  $A$ , it is decidable in time that is cubic in the number of transitions of  $A$ , whether  $A$  is finitely, polynomially, or exponentially ambiguous.

Ravikumar and Ibarra [20] considered the relationship between different types of ambiguity of NFAs to the succinctness of their representations, and they provided a complete picture for unary and bounded languages. Exponentially and polynomially ambiguous NFAs were separated by Leung [15] by providing, for every  $n$ , an exponentially ambiguous  $n$ -state NFA such that every equivalent polynomially ambiguous NFA requires  $2^n - 1$  states.

The UFA-to-DFA tradeoff was improved to the optimal bound  $2^n$  by Leung [16]. He described, for every  $n$ , a binary  $n$ -state UFA with a unique initial state whose equivalent DFA requires  $2^n$  states. A similar binary example with multiple initial states was given by Leiss [14], and a ternary one was presented already by Lupanov [17]; note that the reverse of Lupanov's ternary witness for NFA-to-DFA conversion is deterministic. Leung [16] elaborated Schmidt's lower bound method for the number of states in a UFA. He considered, for a language  $L$ , a matrix whose rows are indexed by strings  $x_i$  and columns by strings  $y_i$ , and the entry in row  $x_i$  and column  $y_j$  is 1 if  $x_i y_j \in L$  and it is 0 otherwise. He showed that the rank of such a matrix provides a lower bound on the number of states in any UFA for  $L$ . Using this method, he was able to describe for every  $n$  an  $n$ -state finitely ambiguous NFA, whose equivalent UFA requires  $2^n - 1$  states.

A lower bound method was further elaborated by Hromkovič et al. [11]. They used communication complexity to show that so-called exact cover of all 1's with monochromatic sub-matrices in a communication matrix of a language provides a lower bound on the size of any UFA for the language. This allowed them to simplify proofs presented in [21, 23]. Using communication complexity methods, Hromkovič and Schnitger [10] showed a separation of finitely and polynomially ambiguous NFAs, and even proved a hierarchy for polynomial ambiguity.

A survey paper on unambiguity in automata theory was presented by Colcombet [6], where he considered word automata, tropical automata, infinite tree automata, and register automata. He showed that the notion of unambiguity is not well understood so far, and that some challenging problems, including complementation of UFAs, remain open.

Unary unambiguous automata were examined by Okhotin [19], who proved that the tight upper bound for UFA-to-DFA conversion in the unary case is given by a function in  $e^{\Theta(\sqrt[3]{n(\ln n)^2})}$ , while the trade-off for NFA-to-UFA conversion is  $e^{\sqrt{n \ln n}(1+o(1))}$ . He also considered the operations of star, concatenation, and complementation on unary UFA languages, and obtained the tight upper bound  $(n-1)^2 + 1$  for star, an upper bound  $mn$  for concatenation which is tight if  $m, n$  are relatively prime, and a lower bound  $n^{2-\epsilon}$  for complementation.

In this paper, we continue this research and study the complexity of basic regular operations on languages represented by unambiguous finite automata. First, we restate the lower bound method from [16,21]. Using the notions of reachable and so-called co-reachable states in an NFA  $N$ , we assign a matrix  $M_N$  to the NFA  $N$  in such a way that the rank of  $M_N$  provides a lower bound on the number of states in any UFA for the language  $L(N)$ . We use this to get all our lower bounds. To get upper bounds, we first construct an NFA for the language resulting from an operation, and then we apply the (incomplete) subset construction to this NFA to get an incomplete DFA, so also UFA, for the resulting language.

## 2 Preliminaries

We assume that the reader is familiar with basic notions in formal languages and automata theory. For details, the reader may refer to [22].

A *nondeterministic finite automaton* (NFA) is a 5-tuple  $N = (Q, \Sigma, \Delta, I, F)$ , where  $Q$  is a finite nonempty set of states,  $\Sigma$  is a finite nonempty input alphabet,  $\Delta \subseteq Q \times \Sigma \times Q$  is the transition relation,  $I \subseteq Q$  is the set of initial states, and  $F \subseteq Q$  is the set of final states. Each element  $(p, a, q)$  of  $\Delta$  is called a *transition* of  $N$ . A *computation* of  $N$  on an input string  $a_1 \cdots a_n$  is a sequence of transitions  $(q_0, a_1, q_1)(q_1, a_2, q_2) \cdots (q_{n-1}, a_n, q_n) \in \Delta^*$ . The computation is *accepting* if  $q_0 \in I$  and  $q_n \in F$ ; in such a case we say that the string  $a_1 \cdots a_n$  is accepted by  $N$ . The *language accepted* by the NFA  $N$  is the set of strings  $L(N) = \{w \in \Sigma^* \mid w \text{ is accepted by } N\}$ .

An NFA  $N = (Q, \Sigma, \Delta, I, F)$  is *unambiguous* (UFA) if it has at most one accepting computation on every input string, and it is *deterministic* (DFA) if  $|I| = 1$  and for each state  $p$  in  $Q$  and each symbol  $a$  in  $\Sigma$ , there is at most one state  $q$  in  $Q$  such that  $(p, a, q)$  is a transition of  $N$ . Let us emphasize that we allow NFAs to have multiple initial states, and DFAs to be incomplete.

The transition relation  $\Delta$  may be viewed as a function from  $Q \times \Sigma$  to  $2^Q$ , which can be extended to the domain  $2^Q \times \Sigma^*$  in the natural way. We denote this function by  $\cdot$ . Using this notation we get  $L(N) = \{w \in \Sigma^* \mid I \cdot w \cap F \neq \emptyset\}$ .

Every NFA  $N = (Q, \Sigma, \cdot, I, F)$  can be converted to an equivalent incomplete DFA  $N' = (2^Q \setminus \{\emptyset\}, \Sigma, \cdot', I, F')$ , where  $F' = \{R \in 2^Q \setminus \{\emptyset\} \mid R \cap F \neq \emptyset\}$ , and for each  $R$  in  $2^Q \setminus \{\emptyset\}$  and each  $a$  in  $\Sigma$ , the partial transition function  $\cdot'$  is defined as follows:  $R \cdot' a = R \cdot a$  if  $R \cdot a \neq \emptyset$  and  $R \cdot' a$  is undefined otherwise. We call the DFA  $N'$  the *incomplete subset automaton* of NFA  $N$ . Since every incomplete DFA is a UFA, we get the following observation.

**Proposition 1.** *If a language  $L$  is accepted by an  $n$ -state NFA, then  $L$  is accepted by a UFA of at most  $2^n - 1$  states. □*

The *reverse*  $w^R$  of a string  $w$  is defined by  $\varepsilon^R = \varepsilon$  and  $(va)^R = av^R$  where  $a \in \Sigma$  and  $v \in \Sigma^*$ . The *reverse of a language*  $L$  is the language  $L^R$  defined by  $L^R = \{w^R \mid w \in L\}$ . The *reverse of an automaton*  $N = (Q, \Sigma, \cdot, I, F)$  is the NFA  $N^R$  obtained from  $N$  by swapping the role of initial and final states and

by reversing all the transitions. Formally, we have  $N^R = (Q, \Sigma, \cdot^R, F, I)$ , where  $q \cdot^R a = \{p \in Q \mid q \in p \cdot a\}$  for each state  $q$  in  $Q$  and each symbol  $a$  in  $\Sigma$ . The NFA  $N^R$  accepts the reverse of the language  $L(N)$ .

Let  $N = (Q, \Sigma, \cdot, I, F)$  be an NFA. We say that a set  $S$  is *reachable* in  $N$  if there is a string  $w$  in  $\Sigma^*$  such that  $S = I \cdot w$ . Next, we say that a set  $T$  is *co-reachable* in  $N$  if  $T$  is reachable in  $N^R$ . In what follows we are interested in *non-empty* reachable and co-reachable sets, and we use the following notation:

$$\mathcal{R} = \{S \subseteq Q \mid S \text{ is reachable in } N \text{ and } S \neq \emptyset\}, \tag{1}$$

$$\mathcal{C} = \{T \subseteq Q \mid T \text{ is co-reachable in } N \text{ and } T \neq \emptyset\}. \tag{2}$$

The next observation uses the notions of reachable and co-reachable sets in an NFA to get a characterization of unambiguous automata.

**Proposition 2.** *Let  $\mathcal{R}$  and  $\mathcal{C}$  be the families of non-empty reachable and co-reachable sets in an NFA  $N$ . Then  $N$  is unambiguous if and only if  $|S \cap T| \leq 1$  for each  $S$  in  $\mathcal{R}$  and each  $T$  in  $\mathcal{C}$ .  $\square$*

If  $N^R$  is deterministic, then each co-reachable set in  $N$  is of size one, and we get the following result.

**Corollary 3.** *Let  $N$  be an NFA. If  $N^R$  is deterministic, then  $N$  is unambiguous.*

Recall that the *state complexity* of a regular language  $L$ ,  $sc(L)$ , is the smallest number of states in any complete DFA accepting the language  $L$ . The state complexity of a regular operation is the maximal state complexity of languages resulting from the operation, considered as the function of state complexities of the arguments. The *nondeterministic state complexity* of languages and operations is defined analogously using NFA representation of languages. We define the *unambiguous state complexity* of a regular language  $L$ ,  $usc(L)$ , as the smallest number of states in any UFA for  $L$ .

To prove that a DFA is minimal, we only need to show that all its states are reachable from the initial state, and that no two distinct states are equivalent. To prove minimality of NFAs, a fooling set lower bound method may be used [2, 8]. To prove a lower bound for the size of a UFA, a method based on ranks of certain matrices was developed by Schmidt [21, Theorem 3.9], Leung [16, Theorem 2] and Hromkovič et al. [11]. We use it in the following statement.

**Proposition 4** ([11, 16, 21]). *Let  $L$  be accepted by an NFA  $N$ . Let  $\mathcal{R}$  and  $\mathcal{C}$  be the families of non-empty reachable and co-reachable sets in  $N$ , respectively. Let  $M_N$  be the matrix in which the rows are indexed by sets in  $\mathcal{R}$ , the columns are indexed by sets in  $\mathcal{C}$ , and in the entry  $(S, T)$ , we have 0/1 if  $S$  and  $T$  are/are not disjoint. Then  $usc(L) \geq \text{rank}(M_N)$ .  $\square$*

*Proof.* Let  $A$  be a minimal UFA accepting  $L$ . Consider a matrix  $M'_A$ , in which rows are indexed by the states of  $A$  and columns are indexed by strings generating the co-reachable sets in  $\mathcal{C}$ . The entry  $(q, w)$  is 1 if  $w^R$  is accepted by  $A$  from  $q$ , and it is 0 otherwise. Then every row of  $M_N$  is a sum of the rows of  $M'_A$  corresponding

to the states in  $S$ : Notice that since  $A$  is a UFA, for every column there is at most one such row that contains a 1. Thus every row of  $M_N$  is a linear combination of rows in  $M'_A$ , and therefore  $\text{rank}(M_N) \leq \text{rank}(M'_A) \leq \text{usc}(L)$ .  $\square$

Throughout our paper, we use the following observation from [15] and its corollary stated in the proposition below.

**Lemma 5 ([15, Lemma 3]).** *Let  $|Q| = n$  and  $M_n$  be a  $2^n - 1 \times 2^n - 1$  matrix over the field with characteristic 2 with rows and columns indexed by a non-empty subsets of  $Q$  such that  $M_n(S, T) = 1$  if  $S \cap T \neq \emptyset$  and  $M_n(S, T) = 0$  otherwise. Then the rank of  $M_n$  is  $2^n - 1$ .*  $\square$

**Proposition 6.** *Let  $L$  be accepted by an NFA  $N$ . Let  $\mathcal{R}$  be the family of all non-empty reachable sets in  $N$ . If each non-empty set is co-reachable in NFA  $N$ , then  $\text{usc}(L) \geq |\mathcal{R}|$ .*  $\square$

*Proof.* Let  $N = (Q, \Sigma, \cdot, I, F)$  be an NFA for  $L$  with  $|Q| = n$ . Consider the matrix  $M_N$  given by Proposition 4. Notice that  $M_N$  contains  $|\mathcal{R}|$  rows of the matrix  $M_n$  given in Lemma 5. By Lemma 5, the rank of  $M_n$  is  $2^n - 1$ , so the rows of  $M_n$  are linearly independent. Therefore all the rows of  $M_N$  must be linearly independent, and we have  $\text{rank}(M_N) = |\mathcal{R}|$ . Hence  $\text{usc}(L) \geq \text{rank}(M_N) = |\mathcal{R}|$  by Proposition 4.  $\square$

### 3 Operations on Unambiguous Finite Automata

We start with the reversal and intersection operations. Then we continue with left and right quotients. Notice that if an NFA  $N$  is unambiguous then  $N^R$  is also unambiguous. Hence we get the following result.

**Theorem 7 (Reversal).** *Let  $L$  be a regular language. Then  $\text{usc}(L^R) = \text{usc}(L)$ .*

**Theorem 8 (Intersection).** *Let  $K$  and  $L$  be languages over  $\Sigma$  with  $\text{usc}(K) = m$  and  $\text{usc}(L) = n$ . Then  $\text{usc}(K \cap L) \leq mn$ , and the bound is tight if  $|\Sigma| \geq 2$ .*

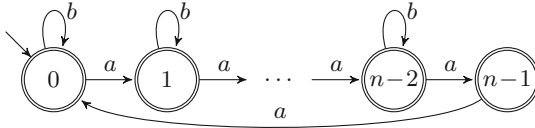
The *left quotient* of a language  $L$  by a string  $w$  is  $w \setminus L = \{x \mid wx \in L\}$ , and the left quotient of a language  $L$  by a language  $K$  is the language  $K \setminus L = \bigcup_{w \in K} w \setminus L$ . The state complexity of the left quotient operation is  $2^n - 1$  [25], and its nondeterministic state complexity is  $n + 1$  [13]. In both cases, the witness languages are defined over a binary alphabet. Our next result shows that the tight upper bound for UFAs is  $2^n - 1$ . To prove tightness we use a binary alphabet.

The *right quotient* of a language  $L$  by a string  $w$  is  $L/w = \{x \mid xw \in L\}$ , and the right quotient of a language  $L$  by a language  $K$  is  $L/K = \bigcup_{w \in K} L/w$ . If a language  $L$  is accepted by an  $n$ -state DFA or NFA  $A$ , then the language  $L/K$  is accepted by an automaton that is exactly the same as  $A$ , except for the set of final states that consists of all states of  $A$ , from which some string in  $K$  is accepted by  $A$  [25]. Thus  $\text{sc}(L/K) \leq n$  and  $\text{nsc}(L/K) \leq n$ . The tightness of the first upper bound has been shown using binary languages in [25]. The second upper bound is met by unary languages  $a^{\geq m-1}$  and  $a^{\leq n-1}$ . Our next aim is to show that the tight upper bound for unambiguous finite automata is  $2^n - 1$ , with witnesses defined over a binary alphabet.

**Theorem 9 (Left and Right Quotient).** *Let  $K, L \subseteq \Sigma^*$ ,  $\text{usc}(K) = m$ , and  $\text{usc}(L) = n$ . Then*

- (a)  $\text{usc}(K \setminus L) \leq 2^n - 1$ , and the bound is tight if  $|\Sigma| \geq 2$ ;
- (b)  $\text{usc}(L / K) \leq 2^n - 1$ , and the bound is tight if  $|\Sigma| \geq 2$ .

*Proof.* (a) To get an upper bound, let  $A$  be an  $n$ -state UFA for  $L$ . Construct an  $n$ -state NFA  $N$  for  $K \setminus L$  from  $A$  by making initial all states of  $A$  that are reachable from the initial set by some string in  $K$ . By Proposition 1,  $\text{usc}(K \setminus L) \leq 2^n - 1$ .

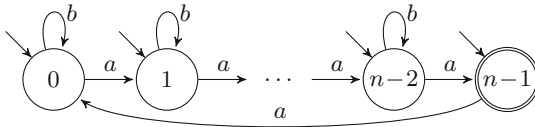


**Fig. 1.** The UFA of a language  $L$  with  $\text{usc}(K \setminus L) = 2^n - 1$ , where  $K = a^{\geq m-1}$ .

For tightness, let  $K = \{a^k \mid k \geq m - 1\}$  and  $L$  be the language accepted by the  $n$ -state DFA  $A = (\{0, 1, \dots, n - 1\}, \{a, b\}, \{0\}, \{0, 1, \dots, n - 1\})$  shown in Fig. 1. Notice that each state of  $A$  is reachable by some string in  $K$ . Construct an  $n$ -state NFA  $N$  for  $K \setminus L$  from  $A$  by making all the states initial. Hence the initial set of  $N$  is  $\{0, 1, \dots, n - 1\}$ . Next, we can shift every reachable subset right by one (modulo  $n$ ) by reading  $a$ , and we can remove the state  $n$  from any subset containing state  $n$  by reading  $b$ . Therefore each non-empty set is reachable in  $N$ .

To construct  $N^R$ , we only need to reverse the transitions on  $a$  in  $N$ . The initial subset of  $N^R$  is  $\{0, 1, \dots, n - 1\}$ , and we can again shift any subset and remove one state as before. It follows that every non-empty set is reachable in  $N^R$ , that is, co-reachable in NFA  $N$ . By Proposition 6, we have  $\text{usc}(K \setminus L) \geq 2^n - 1$ .

(b) To get an upper bound, let  $A$  be an  $n$ -state UFA for  $L$ . Construct an  $n$ -state NFA for  $L / K$  as described above. By Proposition 1,  $\text{usc}(L / K) \leq 2^n - 1$ .



**Fig. 2.** The UFA of a language  $L$  with  $\text{usc}(L / K) = 2^n - 1$ , where  $K = a^{\geq m-1}$ .

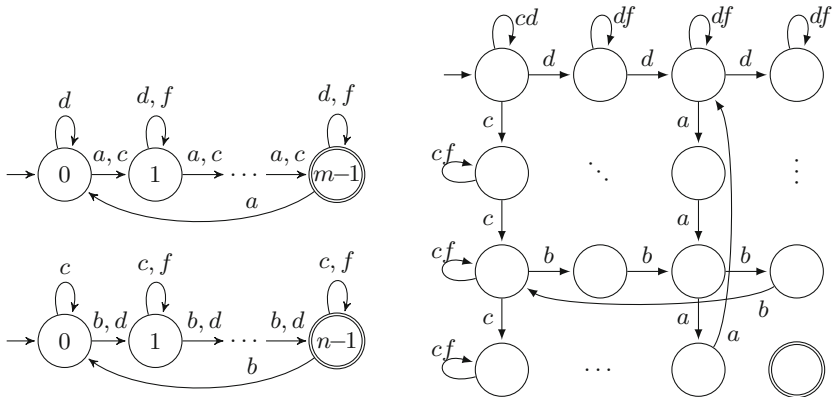
To prove tightness, let  $K = \{a^k \mid k \geq m - 1\}$  and  $L$  be the language accepted by the  $n$ -state NFA  $A = (\{0, 1, \dots, n - 1\}, \{a, b\}, \{0, 1, \dots, n - 1\}, \{n - 1\})$  shown in Fig. 2. Since the automaton  $A^R$  is deterministic, the NFA  $A$  is unambiguous by Corollary 3. Since a string in  $K$  is accepted by  $A$  from each state of  $A$ , we construct an NFA  $N$  for  $L / K$  from  $A$  by making all the states of  $A$  final. Notice that we obtain the same NFA as in the proof of the previous lemma, thus by the same arguments  $\text{usc}(L / K) \geq 2^n - 1$ . □

Now let us continue with the shuffle and concatenation operations. The *shuffle* of two strings  $u$  and  $v$  over an alphabet  $\Sigma$  is defined as the set of strings  $u \sqcup v = \{u_1v_1 \cdots u_kv_k \mid u = u_1 \cdots u_k, v = v_1 \cdots v_k, u_1, \dots, u_k, v_1, \dots, v_k \in \Sigma^*\}$ . The shuffle of languages  $K$  and  $L$  over  $\Sigma$  is defined by  $K \sqcup L = \bigcup_{u \in K, v \in L} u \sqcup v$ . The state complexity of the shuffle operation on languages represented by incomplete deterministic automata was studied by Cămpeanu et al. [3]. They proved that  $2^{mn} - 1$  is a tight upper bound for that case. Here we show that the same upper bound is tight also for UFAs, and to prove tightness, we use almost the same languages as in [3, Theorem 1]. To the best of our knowledge, the problem is still open for complete deterministic automata.

**Theorem 10 (Shuffle).** *Let  $K, L \subseteq \Sigma^*$ ,  $\text{usc}(K) = m$ , and  $\text{usc}(L) = n$ . Then  $\text{usc}(L \sqcup K) \leq 2^{mn} - 1$ , and the bound is tight if  $|\Sigma| \geq 5$ .*

*Proof.* Let  $A = (Q_A, \Sigma, \cdot_A, I_A, F_A)$  and  $B = (Q_B, \Sigma, \cdot_B, I_B, F_B)$  be  $m$ - and  $n$ -state UFAs for  $K$  and  $L$  respectively. Then  $K \sqcup L$  is accepted by an  $mn$ -state NFA  $N = (Q_A \times Q_B, \Sigma, \cdot, I_A \times I_B, F_A \times F_B)$ , where for each state  $(p, q)$  in  $Q_A \times Q_B$  and each symbol  $a$  in  $\Sigma$ , we have  $(p, q) \cdot a = (p \cdot_A a \times \{q\}) \cup (\{p\} \times q \cdot_B a)$ . Hence  $\text{usc}(K \sqcup L) \leq 2^{mn} - 1$  by Proposition 1.

To prove tightness, let  $\Sigma = \{a, b, c, d, f\}$ . Let  $K$  and  $L$  be the regular languages accepted by DFAs  $A = (\{0, 1, \dots, m-1\}, \Sigma, \cdot_A, \{0\}, \{m-1\})$  and  $B = (\{0, 1, \dots, n-1\}, \Sigma, \cdot_B, \{0\}, \{n-1\})$  shown in Fig. 3(left); notice that these DFAs are the same as in [3, Theorem 1] up to the position of final states. Construct an NFA  $N$  for  $K \sqcup L$  as described above. Figure 3(right) shows a sketch of the resulting NFA. It is shown in [3] that each non-empty set is reachable in  $N$ : The initial set  $\{(0, 0)\}$  goes to the full set  $\{0, 1, \dots, m-1\} \times \{0, 1, \dots, n-1\}$  by  $c^m d^n$ , and for each subset  $S$  with  $(i, j) \in S$ , we have  $S \cdot a^{m-i} b^{n-j} f a^i b^j = S \setminus \{(i, j)\}$ . Next, in  $N^R$  we have  $\{(m-1, n-1)\} \cdot^R c^m d^n = \{0, 1, \dots, m-1\} \times \{0, 1, \dots, n-1\}$ , and  $S \cdot^R a^i b^j f a^{m-i} b^{n-j} = S \setminus \{(i, j)\}$  for each subset  $S$  with  $(i, j) \in S$ . It follows that each non-empty set is co-reachable in  $N$ , so  $\text{usc}(L) \geq 2^{mn} - 1$ .  $\square$



**Fig. 3.** Witness UFAs for shuffle (left) and a sketch of the resulting NFA  $N$ .

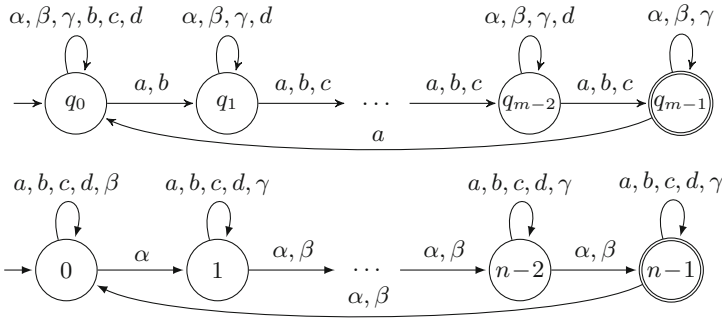
The *concatenation* of languages  $K$  and  $L$  is  $KL = \{uv \mid u \in K \text{ and } v \in L\}$ . The state complexity of concatenation is  $m2^n - 2^{n-1}$ , and its nondeterministic state complexity is  $m + n$ . In both cases, the witnesses are defined over a binary alphabet [9, 13, 18, 25]. In the next theorem we get a tight upper bound for concatenation on UFAs. To prove tightness, we use a seven-letter alphabet.

**Theorem 11 (Concatenation).** *Let  $K, L \subseteq \Sigma^*$ ,  $\text{usc}(K) = m$ , and  $\text{usc}(L) = n$ , where  $m, n \geq 2$ . Then  $\text{usc}(KL) \leq \frac{3}{4} \cdot 2^{m+n} - 1$ , and the bound is tight if  $|\Sigma| \geq 7$ .*

*Proof.* Let  $A = (Q_A, \Sigma, \cdot_A, I_A, F_A)$  and  $B = (Q_B, \Sigma, \cdot_B, I_B, F_B)$  be UFAs for languages  $K$  and  $L$ , respectively. Let  $|Q_A| = m$ ,  $|F_A| = k$ ,  $|Q_B| = n$ ,  $|I_B| = \ell$ . Construct an NFA  $N = (Q_A \cup Q_B, \Sigma, \cdot, I, F_B)$  for  $KL$ , where for each  $q$  in  $Q_A \cup Q_B$  and each  $a$  in  $\Sigma$ ,

$$q \cdot a = \begin{cases} q \cdot_A a, & \text{if } q \in Q_A \text{ and } q \cdot_A a \cap F_A = \emptyset; \\ q \cdot_A a \cup I_B, & \text{if } q \in Q_A \text{ and } q \cdot_A a \cap F_A \neq \emptyset; \\ q \cdot_B a, & \text{if } q \in Q_B, \end{cases}$$

and  $I = I_A$  if  $I_A \cap F_A = \emptyset$  and  $I = I_A \cup I_B$  otherwise. Notice that if a set  $S$  is reachable in the NFA  $N$  and  $S \cap F_A \neq \emptyset$ , then  $I_B \subseteq S$ . It follows that the number of reachable sets is  $2^{m-k}2^n + (2^m - 2^{m-k})2^{n-\ell}$ , which is maximal if  $\ell = 1$ . In such a case, this number equals  $(2^m + 2^{m-k})2^{n-1}$ , which is maximal if  $k = 1$ . After excluding the empty set, we get the upper bound.



**Fig. 4.** Witness UFAs for concatenation meeting the upper bound  $\frac{3}{4} \cdot 2^{m+n} - 1$ .

For tightness, we can use  $K$  and  $L$  over  $\{a, b, c, d, \alpha, \beta, \gamma\}$  accepted by automata  $A$  and  $B$  shown in Fig. 4, where  $Q_A = \{q_0, q_1, \dots, q_{m-1}\}$  and  $Q_B = \{0, 1, \dots, n-1\}$ . Notice that  $A^R$  and  $B$  are deterministic.  $\square$

Now we consider the Kleene closure (star) and positive closure operations. For a language  $L$ , the *star* of  $L$  is the language  $L^* = \bigcup_{i \geq 0} L^i$ , where  $L^0 = \{\varepsilon\}$  and  $L^{i+1} = L^i L$ . The *positive closure* of  $L$  is  $L^+ = \bigcup_{i \geq 1} L^i$ . The state complexity of



the star operation is  $\frac{3}{4} \cdot 2^n$  with binary witness languages [18, 25]. In the unary case, the tight upper bound is  $(n - 1)^2 + 1$  [4, 25]. The nondeterministic state complexity of star is  $n + 1$ , with witnesses defined over a unary alphabet [9].

**Theorem 12 (Positive Closure and Star).** *Let  $L$  be a language over  $\Sigma$  with  $\text{usc}(L) = n$ , where  $n \geq 2$ . Then*

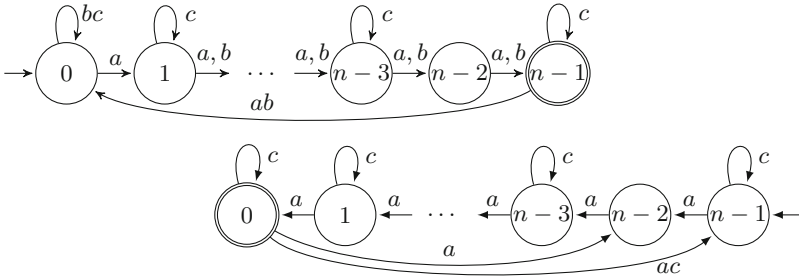
- (a)  $\text{usc}(L^+) \leq \frac{3}{4} \cdot 2^n - 1$ , and the bound is tight if  $|\Sigma| \geq 3$ ;
- (b)  $\text{usc}(L^*) \leq \frac{3}{4} \cdot 2^n$ , and the bound is tight if  $|\Sigma| \geq 3$ .

*Proof.* (a) To get an upper bound, let  $A = (Q, \Sigma, \cdot, I, F)$  be an  $n$ -state UFA for  $L$ . Construct an NFA  $N = (Q, \Sigma, \cdot^+, I, F)$  for  $L^+$  where the transition function  $\cdot^+$  is defined as

$$q \cdot^+ a = \begin{cases} q \cdot a \cup I, & \text{if } q \cdot a \cap F \neq \emptyset; \\ q \cdot a, & \text{otherwise} \end{cases}$$

for each state  $q$  in  $Q$  and each symbol  $a$  in  $\Sigma$ . Notice that if a set  $S$  is reachable in  $N$  and  $S \cap F \neq \emptyset$ , then  $I \subseteq S$ . We can show that there are at most  $\frac{3}{4} \cdot 2^n - 1$  reachable non-empty subsets in  $N$ . This proves the upper bound.

To prove tightness, let  $L$  be the language accepted by the ternary DFA  $A$  shown in Fig. 5(top). Construct the NFA  $N$  for  $L^+$  as described above. Notice that the DFA  $A$  restricted to the alphabet  $\{a, b\}$  is the same as the witness DFA for the star operation from [25, Theorem 3.3, Fig. 4] In particular, this means that  $N$  has  $\frac{3}{4} \cdot 2^n - 1$  non-empty reachable subsets. We can show that each non-empty set is co-reachable in  $N$ .



**Fig. 5.** The witness UFA for positive closure meeting the upper bound  $\frac{3}{4} \cdot 2^n - 1$  (top), and the transitions on  $a, c$  in the NFA  $N^R$  (bottom).

(b) The upper bound follows from the case (a) since if  $\varepsilon \in L$ , then  $L^+ = L^*$ , and otherwise we only need to add one more initial and final state to the UFA for  $L^+$  to accept the empty string. The resulting automaton is unambiguous since the new state accepts only the empty string which is not accepted by UFA for  $L^+$ . For tightness, consider the language  $L$  accepted by the UFA  $A$  shown in Fig. 5(top). Construct an NFA  $N$  for  $L^*$  from UFA  $A$  by adding a new initial

and final state  $q_0$ , and by adding the transitions on  $a, b$  from  $n - 2$  to 0, and the transition by  $c$  from  $n - 1$  to 0. As shown in [25, Theorem 3.3] the NFA  $N$  has  $\frac{3}{4} \cdot 2^n$  reachable sets: the initial set  $\{q_0, 0\}$ , all the subsets of  $\{0, 1, \dots, n - 1\}$  containing state 0, and all the non-empty subsets of  $\{1, 2, \dots, n - 2\}$ . Consider the NFA  $N^R$ . The initial set of  $N^R$  is  $\{q_0, n - 1\}$ . Next, as we have shown that each non-empty set is reachable in  $N^R$ . Now consider the  $\frac{3}{4} \cdot 2^n \times 2^n$  matrix  $M_N$ , and show that its rank is  $\frac{3}{4} \cdot 2^n \times 2^n$ .  $\square$

### 4 Partial Results for Complementation and Union

In this section we present partial results for the complementation and union operations on UFA languages. The complement of a language  $L$  over  $\Sigma$  is the language  $L^c = \Sigma^* \setminus L$ . A language and its complement have the same state complexity since to get a DFA for the complement of  $L$ , we only need to interchange the sets of final and non-final states in a DFA for  $L$ . For NFAs, the tight upper bound for complementation is  $2^n$  with witnesses defined over a binary alphabet [9, 13]. For unary UFAs, the problem was studied by Okhotin, who provided a lower bound  $n^{2-o(1)}$  for complementation of unary UFAs [19, Theorem 6]. In the next theorem we deal with an upper bound. Then we consider union.

**Theorem 13 (Complementation: Upper Bound).** *Let  $L$  be a regular language with  $\text{usc}(L) = n$ , where  $n \geq 7$ . Then  $\text{usc}(L^c) \leq 2^{0.79n + \log n}$ .*

*Proof.* Let  $A$  be an  $n$ -state UFA for  $L$  and  $\mathcal{R}$  and  $\mathcal{C}$  be the sets of non-empty reachable and co-reachable sets of  $A$ . First, we show that  $\text{usc}(L^c) \leq \min\{|\mathcal{R}|, |\mathcal{C}|\}$ . We have  $\text{usc}(L^c) \leq |\mathcal{R}|$  since we can get a DFA for  $L^c$  by applying the subset construction to  $A$  and by interchanging the sets of final and non-final states in the resulting DFA that has  $|\mathcal{R}|$  reachable states. Next, we have  $\text{usc}(L^c) \leq |\mathcal{C}|$  since the NFA  $A^R$  is unambiguous, so  $\text{usc}((L^R)^c) \leq |\mathcal{C}|$  which means that  $\text{usc}(L^c) \leq |\mathcal{C}|$  since complement and reversal commutes and the reverse of a UFA is a UFA.

Next, let  $k = \max\{|X| \mid X \in \mathcal{R}\}$ , and pick a set  $S$  in  $\mathcal{R}$  of size  $k$ . Then each set in  $\mathcal{R}$  has size at most  $k$ , and each set in  $\mathcal{C}$  may have at most one element in  $S$  by Proposition 2. Thus

$$|\mathcal{R}| \leq \binom{n}{1} + \binom{n}{2} + \dots + \binom{n}{k} \text{ and } |\mathcal{C}| \leq (k + 1)2^{n-k}.$$

If  $k \geq n/2$ , then  $|\mathcal{C}| \leq (n/2 + 1) \cdot 2^{n/2} \leq 2^{0.5n + \log n}$ , and the theorem follows. Now assume that  $k < n/2$ . Then  $|\mathcal{R}| \leq k \binom{n}{k} \leq n \left(\frac{en}{k}\right)^k$  and  $|\mathcal{C}| \leq n2^{n-k}$ . Let  $r(k) = n \left(\frac{en}{k}\right)^k$  and  $c(k) = n2^{n-k}$ . Then  $r(k)$  increases, while  $c(k)$  decreases with  $k$ . It follows that if we pick a  $k_0$  such that  $k_0 < n/2$ , then  $\text{usc}(L^c) \leq r(k_0)$  if  $k \leq k_0$ , and  $\text{usc}(L^c) \leq c(k_0)$  otherwise. By setting  $k = nx$  and by solving  $\left(\frac{en}{nx}\right)^{nx} = 2^{n-nx}$ , we get  $x_0 = 0.2144$ ,  $k_0 = 0.2144n$ ,  $r(k_0) \leq 2^{0.7856n + \log n}$ , and  $c(k_0) \leq 2^{0.785629n + \log n}$ . This completes our proof.  $\square$

**Proposition 14 (Union).** *Let  $K$  and  $L$  be languages over  $\Sigma$  with  $\text{usc}(K) = m$  and  $\text{usc}(L) = n$ , where  $1 \leq m \leq n$ . Then*

- (a)  $\text{usc}(K \cup L) \leq m + n \cdot \text{usc}(K^c) \leq m + n2^{0.79n + \log n}$ ;  
 (b) *the bound  $mn + m + n$  is met if  $|\Sigma| \geq 4$ .*

*Proof.* (a) The claim follows from the equality  $K \cup L = K \dot{\cup} (L \cap K^c)$ , where  $\dot{\cup}$  denotes a disjoint union, since we have  $\text{usc}(L \cap K^c) \leq n \cdot \text{usc}(L^c)$  by Theorem 8, and, moreover, the NFA for a disjoint union of UFAs is unambiguous. The second inequality is given by Theorem 13. We can prove the lower bound in (b) using a four-letter alphabet.  $\square$

## 5 Conclusions

We investigated the complexity of operations on unambiguous finite automata. Since the reverse of an unambiguous automaton is unambiguous, a language and its reversal have the same complexity for UFAs. Next, we got tight upper bounds for intersection ( $mn$ ), left and right quotients ( $2^n - 1$ ), positive closure ( $\frac{3}{4} \cdot 2^n - 1$ ), star ( $\frac{3}{4} \cdot 2^n$ ), shuffle ( $2^{mn} - 1$ ), and concatenation ( $\frac{3}{4} \cdot 2^{m+n} - 1$ ).

To get upper bounds, we constructed an NFA for the language resulting from an operation, and applied the (incomplete) subset construction to it. For lower bounds, we defined witness languages in such a way that we were able to assign a matrix to a resulting language. The rank of this matrix provided a lower bound on the unambiguous state complexity of the resulting language. To prove tightness, we used a binary alphabet for intersection and left and right quotients, a ternary alphabet for star and positive closure, a five-letter alphabet for shuffle, and a seven-letter alphabet for concatenation. For complementation and union, we provided upper bounds  $2^{0.79n + \log n}$  and  $m + n2^{0.79n + \log n}$ , respectively. Finally, we got a lower bound  $mn + m + n$  for union.

In the case of complementation, we tried to use a fooling set lower bound method, but we were able to describe a fooling set for the complement of an  $n$ -state UFA language only of size  $n + \log n$ . Moreover, it seems that every such fooling set is of size which is quadratic in  $n$  [7]. Thus the fooling set technique cannot be used to get a larger lower bound. Neither the method based on the rank of matrices can be used here since the matrices of a language and its complement have the same rank, up to one. Therefore to get a larger lower bound for complementation, some other techniques should be developed.<sup>1</sup>

## References

1. Allauzen, C., Mohri, M., Rastogi, A.: General algorithms for testing the ambiguity of finite automata and the double-tape ambiguity of finite-state transducers. *Int. J. Found. Comput. Sci.* **22**(4), 883–904 (2011). <http://dx.doi.org/10.1142/S0129054111008477>

<sup>1</sup> The full version can be found at <http://im.saske.sk/~jiraskov/UFA/ufa.pdf>.

2. Birget, J.: Partial orders on words, minimal elements of regular languages and state complexity. *Theor. Comput. Sci.* **119**(2), 267–291 (1993). [http://dx.doi.org/10.1016/0304-3975\(93\)90160-U](http://dx.doi.org/10.1016/0304-3975(93)90160-U)
3. Câmpeanu, C., Salomaa, K., Yu, S.: Tight lower bound for the state complexity of shuffle of regular languages. *J. Autom. Lang. Comb.* **7**(3), 303–310 (2002)
4. Čevorová, K.: Kleene star on unary regular languages. In: Jurgensen, H., Reis, R. (eds.) DCFS 2013. LNCS, vol. 8031, pp. 277–288. Springer, Heidelberg (2013). [http://dx.doi.org/10.1007/978-3-642-39310-5\\_26](http://dx.doi.org/10.1007/978-3-642-39310-5_26)
5. Chan, T., Ibarra, O.H.: On the finite-valuedness problem for sequential machines. *Theor. Comput. Sci.* **23**, 95–101 (1983). [http://dx.doi.org/10.1016/0304-3975\(88\)90012-6](http://dx.doi.org/10.1016/0304-3975(88)90012-6)
6. Colcombet, T.: Unambiguity in automata theory. In: Shallit, J., Okhotin, A. (eds.) DCFS 2015. LNCS, vol. 9118, pp. 3–18. Springer, Heidelberg (2015). [http://dx.doi.org/10.1007/978-3-319-19225-3\\_1](http://dx.doi.org/10.1007/978-3-319-19225-3_1)
7. Eliáš, P.: Fooling sets for complements of UFAs. Unpublished manuscript (2016)
8. Glaister, I., Shallit, J.: A lower bound technique for the size of non-deterministic finite automata. *Inf. Process. Lett.* **59**(2), 75–77 (1996). [http://dx.doi.org/10.1016/0020-0190\(96\)00095-6](http://dx.doi.org/10.1016/0020-0190(96)00095-6)
9. Holzer, M., Kutrib, M.: Nondeterministic descriptonal complexity of regular languages. *Int. J. Found. Comput. Sci.* **14**(6), 1087–1102 (2003). <http://dx.doi.org/10.1142/S0129054103002199>
10. Hromkovič, J., Schnitger, G.: Ambiguity and communication. *Theory Comput. Syst.* **48**(3), 517–534 (2011). <http://dx.doi.org/10.1007/s00224-010-9277-4>
11. Hromkovič, J., Seibert, S., Karhumäki, J., Klauck, H., Schnitger, G.: Communication complexity method for measuring nondeterminism in finite automata. *Inf. Comput.* **172**(2), 202–217 (2002). <http://dx.doi.org/10.1006/inco.2001.3069>
12. Ibarra, O.H., Ravikumar, B.: On sparseness, ambiguity and other decision problems for acceptors and transducers. In: Monien, B., Vidal-Naquet, G. (eds.) STACS 1986. LNCS, vol. 210, pp. 171–179. Springer, Heidelberg (1986). [http://dx.doi.org/10.1007/3-540-16078-7\\_74](http://dx.doi.org/10.1007/3-540-16078-7_74)
13. Jirásková, G.: State complexity of some operations on binary regular languages. *Theor. Comput. Sci.* **330**(2), 287–298 (2005). <http://dx.doi.org/10.1016/j.tcs.2004.04.011>
14. Leiss, E.L.: Succinct representation of regular languages by boolean automata. *Theor. Comput. Sci.* **13**, 323–330 (1981). [http://dx.doi.org/10.1016/S0304-3975\(81\)80005-9](http://dx.doi.org/10.1016/S0304-3975(81)80005-9)
15. Leung, H.: Separating exponentially ambiguous finite automata from polynomially ambiguous finite automata. *SIAM J. Comput.* **27**(4), 1073–1082 (1998). <http://dx.doi.org/10.1137/S0097539793252092>
16. Leung, H.: Descriptonal complexity of NFA of different ambiguity. *Int. J. Found. Comput. Sci.* **16**(5), 975–984 (2005). <http://dx.doi.org/10.1142/S0129054105003418>
17. Lupanov, O.B.: A comparison of two types of finite automata. *Problemy Kibernetiki* 9, *Kibernetiki* (1963), (in Russian) German translation: Über den Vergleich zweier Typen endlicher Quellen. *Probleme der Kybernetik* **6**, 328–335 (1966)
18. Maslov, A.N.: Estimates of the number of states of finite automata. *Soviet Math. Doklady* **11**, 1373–375 (1970)
19. Okhotin, A.: Unambiguous finite automata over a unary alphabet. *Inf. Comput.* **212**, 15–36 (2012). <http://dx.doi.org/10.1016/j.ic.2012.01.003>

20. Ravikumar, B., Ibarra, O.H.: Relating the type of ambiguity of finite automata to the succinctness of their representation. *SIAM J. Comput.* **18**(6), 1263–1282 (1989). <http://dx.doi.org/10.1137/0218083>
21. Schmidt, E.M.: Succinctness of description of context-free, regular, and finite languages. Ph. D. thesis. Cornell University (1978)
22. Sipser, M.: *Introduction to the Theory of Computation*. PWS Publishing Company, Boston (1997)
23. Stearns, R.E., Hunt, H.B.: On the equivalence and containment problems for unambiguous regular expressions, regular grammars and finite automata. *SIAM J. Comput.* **14**(3), 598–611 (1985). <http://dx.doi.org/10.1137/0214044>
24. Weber, A., Seidl, H.: On the degree of ambiguity of finite automata. *Theor. Comput. Sci.* **88**(2), 325–349 (1991). [http://dx.doi.org/10.1016/0304-3975\(91\)90381-B](http://dx.doi.org/10.1016/0304-3975(91)90381-B)
25. Yu, S., Zhuang, Q., Salomaa, K.: The state complexities of some basic operations on regular languages. *Theor. Comput. Sci.* **125**(2), 315–328 (1994). [http://dx.doi.org/10.1016/0304-3975\(92\)00011-F](http://dx.doi.org/10.1016/0304-3975(92)00011-F)