



NFA-to-DFA Trade-Off for Regular Operations

Galina Jirásková, Ivana Krajňáková

► To cite this version:

Galina Jirásková, Ivana Krajňáková. NFA-to-DFA Trade-Off for Regular Operations. 21th International Conference on Descriptive Complexity of Formal Systems (DCFS), Jul 2019, Košice, Slovakia. pp.184-196, 10.1007/978-3-030-23247-4_14 . hal-02387289

HAL Id: hal-02387289

<https://inria.hal.science/hal-02387289>

Submitted on 29 Nov 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

NFA-to-DFA Trade-Off for Regular Operations^{*}

Galina Jirásková and Ivana Krajňáková^(✉)

Mathematical Institute, Slovak Academy of Sciences,
Grešákova 6, 040 01 Košice, Slovakia
jiraskov@saske.sk, krajnakova@saske.sk

Abstract. We examine the operational state complexity assuming that the operands of a regular operation are represented by nondeterministic finite automata, while the language resulting from the operation is required to be represented by a deterministic finite automaton. We get tight upper bounds 2^n for complementation, reversal, and star, 2^m for left and right quotient, 2^{m+n} for union and symmetric difference, $2^{m+n} - 2^m - 2^n + 2$ for intersection, $2^{m+n} - 2^n + 1$ for difference, $\frac{3}{4}2^{m+n}$ for concatenation, and 2^{mn} for shuffle. We use a binary alphabet to describe witnesses for complementation, reversal, star, and left and right quotient, and a quaternary alphabet otherwise. Whenever we use a binary alphabet, it is always optimal.

1 Introduction

The state complexity of a regular language L , $\text{sc}(L)$, is the smallest number of states in any deterministic finite automaton (DFA) recognising L . The state complexity of a k -ary regular operation \circ is a function from \mathbb{N}^k to \mathbb{N} given by

$$(n_1, n_2, \dots, n_k) \mapsto \max\{\text{sc}(\circ(L_1, L_2, \dots, L_k)) \mid \text{sc}(L_i) \leq n_i \text{ for } i = 1, 2, \dots, k\}.$$

The first results on the state complexity of basic regular operations have been obtained by Maslov [11], Birget [1], and Yu et al. [15]. Holzer and Kutrib [6] considered the representation of regular languages by nondeterministic finite automata (NFAs) and defined and studied the nondeterministic state complexity of regular languages and operations in an analogous way. Jirásek Jr. et al. [8, 9] investigated operational state complexity using representation of regular languages by self-verifying and unambiguous finite automata. Notice that in all of the above mentioned cases, the arguments and the results of regular operations are represented by the same computational model.

In this paper, we consider the NFA-to-DFA trade-off for regular operations, that is, we assume that the arguments of an operation are represented by NFAs, while the resulting language is required to be represented by a DFA. Our motivation comes from the following two streams of research.

^{*} Research supported by grant VEGA 2/0132/19 and grant APVV-15-0091. This work was conducted as a part of PhD study of the second author at Comenius University in Bratislava.

While investigating operational state complexity on self-verifying or unambiguous automata, which are nondeterministic, the NFA-to-DFA trade-off provides an upper bound on the complexity of the corresponding operation since every DFA is self-verifying as well as unambiguous. As shown in [8, 9], these upper bounds are tight for several operations.

Our second motivation comes from the research on the state complexity of combined operations that began with the paper by Salomaa et al. [12]. If a combined operation does not contain complementation, we can perform all the included operations using NFAs. Then, the NFA-to-DFA trade-off for the outermost operation can be used to get an upper bound on the desired complexity of a given combined operation.

We examine the NFA-to-DFA trade-off for complementation, intersection, union, difference, symmetric difference, reversal, star, concatenation, shuffle, and left and right quotient. For each of these operations, we get tight upper bound on its NFA-to-DFA trade-off. To describe witnesses, we use either binary or quaternary alphabets. The binary alphabet is always optimal in the sense that the corresponding upper bounds cannot be met by any unary languages.

To conclude this introduction, let us mention that the trade-offs between different models of finite automata have been studied for the forever operator defined as $L \mapsto (\Sigma^* L^c)^c$ by Birget [2] and Hospodár et al. [7].

2 Preliminaries

We assume that the reader is familiar with basic notions in formal languages and automata theory. For details and all unexplained notions, the reader may refer to [14].

Let Σ be a finite non-empty alphabet. Then Σ^* denotes the set of all words over Σ including the empty word ε . If $u, v, w \in \Sigma^*$ and $w = uv$, then u is *prefix* of w . Moreover, if $u \neq w$, then u is a *proper prefix* of w . A *language* over an alphabet Σ is any subset of Σ^* .

If K and L are languages over Σ , then the *complement* of L is $L^c = \Sigma^* \setminus L$. The *intersection*, *union*, *difference*, and *symmetric difference* of K and L are defined as for arbitrary sets. Next, we consider the following regular operations: *concatenation* $KL = \{uv \mid u \in K \text{ and } v \in L\}$, *star* $L^* = \bigcup_{i \geq 0} L^i$ where $L^0 = \{\varepsilon\}$ and $L^{i+1} = L^i L$, *reversal* $L^R = \{w^R \mid w \in L\}$, *shuffle* $K \sqcup L = \{u_1 v_1 u_2 v_2 \cdots u_k v_k \mid u_i, v_i \in \Sigma^*, u_1 u_2 \cdots u_k \in K, v_1 v_2 \cdots v_k \in L\}$, *right quotient* $KL^{-1} = \{x \in \Sigma^* \mid xy \in K \text{ for some } y \in L\}$, and *left quotient* $L^{-1}K = \{x \in \Sigma^* \mid yx \in K \text{ for some } y \in L\}$.

A *nondeterministic finite automaton* (NFA) is a 5-tuple $A = (Q, \Sigma, \cdot, s, F)$, where Q is a finite non-empty set of states, Σ is a finite input alphabet, $s \in Q$ is the *starting* (or *initial*) state, $F \subseteq Q$ is the set of *final* (or *accepting*) states, and $\cdot : Q \times \Sigma \rightarrow 2^Q$ is the transition function which can be extended to the domain $2^Q \times \Sigma^*$ in the natural way. The language recognised by the NFA A is the set of words $L(A) = \{w \in \Sigma^* \mid s \cdot w \cap F \neq \emptyset\}$.

An NFA A is a *deterministic finite automaton* if $|q \cdot a| = 1$ for each $q \in Q$ and $a \in \Sigma$. In such a case we write $q \cdot a = q'$ instead of $q \cdot a = \{q'\}$, and use $q \xrightarrow{a} q'$ to denote that $q \cdot a = q'$. Sometimes we permit non-deterministic automata to have more initial states; in such a case we use an abbreviation NNFA from [14].

A subset S of Q is *reachable* in an NNFA $A = (Q, \Sigma, \cdot, I, F)$ if $S = I \cdot w$ for some $w \in \Sigma^*$, and it is *co-reachable* if it is reachable in the reversed automaton A^R obtained from A by reversing all its transitions and swapping the roles of the initial and final states.

Every NNFA $A = (Q, \Sigma, \cdot, I, F)$ has an equivalent deterministic finite automaton $\mathcal{D}(A) = (2^Q, \Sigma, \cdot, I, \{T \subseteq Q \mid T \cap F \neq \emptyset\})$. The DFA $\mathcal{D}(A)$ is called the *subset automaton* of A . The following observation provides a sufficient condition that guarantees distinguishability of all states in a subset automaton. We use this lemma throughout the paper.

Lemma 1 (Distinguishability). *Let A be an NFA such that for every state q of A the singleton set $\{q\}$ is co-reachable in A . Then every two distinct states of the subset automaton $\mathcal{D}(A)$ are distinguishable.*

Proof. Let us take two distinct subsets S and T of $\mathcal{D}(A)$. Without loss of generality, let $q \in S \setminus T$. Since the set $\{q\}$ is co-reachable in A , there is a word w_q that is accepted by A from the state q and rejected from every other state. It follows that in $\mathcal{D}(A)$, the word w_q is accepted from S and rejected from T . Hence S and T are distinguishable in $\mathcal{D}(A)$. \square

The next lemma shows that every subset of the state set of the NFA from Fig. 1 is reachable in the corresponding subset automaton. It also shows that to reach every non-empty subset, the final state $n - 1$ may be visited only in the very last steps. This is an important property which is used later to get the results for concatenation.

Lemma 2 (Reachability). *Let $A = (\{0, 1, \dots, n - 1\}, \{a, b\}, \cdot, 0, \{n - 1\})$ be the NFA from Fig. 1. where $i \cdot a = \{i + 1\}$ if $0 \leq i \leq n - 2$, and $i \cdot b = \{0, i\}$. Then for each subset S of $\{0, 1, \dots, n - 1\}$, there exists a word $u_S \in \{a, b\}^*$ such that $0 \cdot u_S = S$. Moreover,*

- (1) *if $S \neq \emptyset$ and $n - 1 \notin S$, then $n - 1 \notin 0 \cdot u'$ for each prefix u' of u_S ;*
- (2) *if $n - 1 \in S$ and $0 \notin S$, then $n - 1 \notin 0 \cdot u'$ for each proper prefix u' of u_S ,*
- (3) *if $n - 1 \in S$ and $0 \in S$, then $u_S = u'b$ and $n - 1 \notin 0 \cdot u''$ for each proper prefix u'' of u' ,*

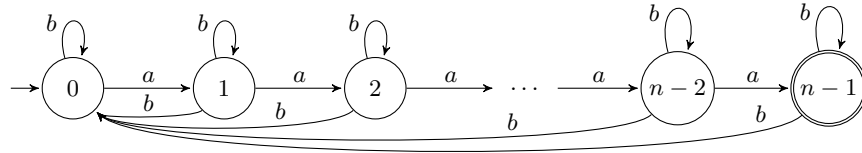


Fig. 1. A binary witness NFA for determinization and complementation.

Proof. In the subset automaton $\mathcal{D}(A)$, each singleton set $\{i\}$ is reached from the initial subset $\{0\}$ by a^i and the empty set is reached from $\{n-1\}$ by a . Each set $\{i_1, i_2, \dots, i_k\}$ of size k , where $2 \leq k \leq n$ and $0 \leq i_1 < i_2 < \dots < i_k \leq n-1$, is reached from the set $\{i_2 - i_1, i_3 - i_1, \dots, i_k - i_1\}$ of size $k-1$ by ba^{i_1} . This proves the reachability of all subsets of $\{0, 1, \dots, n-1\}$ by induction. Hence for each $S \subseteq \{0, 1, \dots, n-1\}$, there is a word $u_S \in \{a, b\}^*$ such that $0 \cdot u_S = S$. By a careful analysis of the proof above, we get the properties (1), (2), and (3). \square

3 Complementation, Reversal, Star, and Concatenation

In this section we examine the NFA-to-DFA trade-off for basic unary operations and concatenation. We start with complementation. Its state complexity is n while its non-deterministic state complexity is 2^n [10]. The next theorem shows that its NFA-to-DFA trade-off is 2^n as well.

Theorem 3 (Complementation). *Let L be a language over Σ recognised by an n -state NFA. Then $\text{sc}(L^c) \leq 2^n$, and the bound is tight if $|\Sigma| \geq 2$.*

Proof. Since $\text{sc}(L^c) = \text{sc}(L)$, the upper bound follows from the upper bound on determinization. For tightness, let L be the language recognised by the NFA A shown in Fig. 1 on page 3. By Lemma 2, every subset of $\{0, 1, \dots, n-1\}$ is reachable in the subset automaton $\mathcal{D}(A)$. Since every singleton set is co-reachable in A via a word in a^* , all states of $\mathcal{D}(A)$ are pairwise distinguishable by Lemma 1. \square

Notice that the binary alphabet used in the previous proof is optimal since every unary n -state NFA can be simulated by a DFA of $2^{O(\sqrt{n \ln n})}$ states as shown by Chrobak [5]. Let us continue with the reversal operation. Note that it is enough to take the reversal of any DFA with one final state meeting the upper bound 2^n on the state complexity of reversal. Such a binary DFA was described by Šebej [13]. Here we describe a different witness with significantly simpler proof.

Theorem 4 (Reversal). *Let L be a language over Σ recognised by an n -state NFA, where $n \geq 2$. Then $\text{sc}(L^R) \leq 2^n$, and the bound is tight if $|\Sigma| \geq 2$.*

Proof. Let L be accepted by an n -state NFA $A = (Q, \Sigma, \cdot, s, F)$. By reversing all the transitions in A and taking F as the set of starting states and $\{s\}$ as set of final states we obtain an n -state NNFA that accepts L^R . It follows that L^R is accepted by a DFA with at most 2^n states.

To prove tightness, consider the binary language L recognised by the n -state NFA $N = (\{0, 1, \dots, n-1\}, \{a, b\}, \cdot, 0, \{0, 1, \dots, n-1\})$ shown in Fig. 2 where $i \cdot a = \{i+1 \bmod n\}$, $i \cdot b = \{i\}$ if $i \geq 1$. By reversing NFA N we get an NNFA N^R that recognises L^R .

The set of initial state of N^R is $\{0, 1, \dots, n-1\}$ and its unique final state is 0. Notice that each subset of the state set of N^R can be shifted cyclically by one by reading a , and the state 0 can be eliminated from every set containing 0

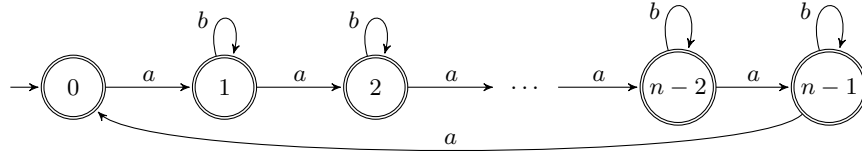


Fig. 2. A binary witness NFA for reversal meeting the upper bound 2^n .

by reading b . It follows that every subset of $\{0, 1, \dots, n-1\}$ can be reached from the initial subset $\{0, 1, \dots, n-1\}$ in the subset automaton $\mathcal{D}(N^R)$. Next, every set $\{i\}$ is co-reachable in N^R via a word in a^* and using Lemma 1 we get that every two distinct states of the $\mathcal{D}(N^R)$ are distinguishable. \square

The binary alphabet used in the previous theorem is optimal for the same reason as in the case of complementation. We continue with the star operation. While its state complexity is $\frac{3}{4}2^n$ [11] and its nondeterministic state complexity is $n+1$ [10], we show that the NFA-to-DFA trade-off for star is 2^n .

Theorem 5 (Star). *Let $n \geq 2$. Let L be a language over an alphabet Σ recognised by an n -state NFA. Then $\text{sc}(L^*) \leq 2^n$, and the bound is tight if $|\Sigma| \geq 2$.*

Proof. Let L be recognised by an n -state NFA $A = (Q, \Sigma, \cdot, s, F)$. Construct an NNFA N recognising L^* from A as follows. First, for each transition (p, a, q) in A with $q \in F$, add the transition (p, a, s) . Next, if $s \notin F$, then add a new initial and final state q_0 to accept the empty word. Consider the subset automaton $\mathcal{D}(N)$. The only reachable set in $\mathcal{D}(N)$ containing the state q_0 is the initial subset $\{s, q_0\}$. All the remaining reachable sets are subsets of Q . Moreover, if a reachable set contains a final state of A , then it also contains the state s . If A has a final state different from s , then at least 2^{n-2} sets are unreachable in $\mathcal{D}(N)$, so the upper bound is $1 + (3/4)2^n$ in this case. If $F = \{s\}$, then the construction above results in the same automaton, so $L^* = L$. In such a case, the upper bound is 2^n .

To prove tightness, consider the binary language L recognised by the n -state NFA $A = (\{0, 1, \dots, n-1\}, \{a, b\}, \cdot, 0, \{0\})$ shown in Fig. 3 where for each state i , $i \cdot a = \{i+1 \bmod n\}$, $i \cdot b = \{0, i\}$ if $i \geq 1$. Then $L^* = L$. In the subset automaton $\mathcal{D}(A)$, the empty set is reached from the initial subset $\{0\}$ by b . The reachability of all non-empty subsets is proved exactly the same way as in the proof of Lemma 2. Since every singleton set is co-reachable in A via a word in a^* , all the states of $\mathcal{D}(A)$ are pairwise distinguishable by Lemma 1. \square

The witness from the previous proof is described over a binary alphabet. It is impossible to meet the upper bound 2^n in the unary case since every unary n -state NFA can be simulated by a DFA with $2^{O(\sqrt{n \ln n})}$ states. The unary language recognised by the NFA from Fig. 4 provides a lower bound $(n-1)^2 + 2$; notice that this NFA is not unambiguous. We conjecture that this lower bound is tight. Our computations support this conjecture.

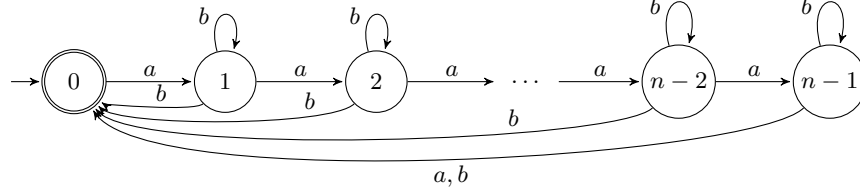


Fig. 3. A binary witness NFA for star meeting the upper bound 2^n .

We conclude this section with the concatenation operation the state complexity of which is $m2^n - 2^{n-1}$ [11] and nondeterministic complexity is $m + n$ [6]. The next theorem shows that NFA-to-DFA trade-off for concatenation is $\frac{3}{4}2^{m+n}$, that is, it is exponential in both m and n .

Theorem 6 (Concatenation). *Let K and L be non-empty languages over an alphabet Σ recognised by an m -state and n -state NFA, respectively, with $m, n \geq 3$. Then $\text{sc}(KL) \leq \frac{3}{4}2^{m+n}$, and the bound is tight if $|\Sigma| \geq 4$.*

Proof. Let $A = (Q_A, \Sigma, \cdot_A, s_A, F_A)$ and $B = (Q_B, \Sigma, \cdot_B, s_B, F_B)$ be NFAs recognising K and L , respectively, with $|Q_A| = m$, $|Q_B| = n$. Construct an NNFA N for KL from NFAs A and B as follows. For each transition (p, a, q) in NFA A with $q \in F_A$, add the transition (p, a, s_B) . The set of initial states of N is $\{s_A\}$ if $s_A \notin F_A$, or $\{s_A, s_B\}$ if $s_A \in F_A$. The set of final states of N is F_B . The following condition holds in the subset automaton $\mathcal{D}(N)$: each reachable subset containing a state from F_A must contain the state s_B . It follows that at least 2^{m+n-2} are unreachable in $\mathcal{D}(N)$, and the upper bound follows.

For tightness, let $\Sigma = \{a, b, c, d\}$ and K and L be the languages over Σ recognised by the NFAs $A = (\{q_0, q_1, \dots, q_{m-1}\}, \Sigma, \cdot_A, q_0, \{q_{m-1}\})$ and $B = (\{0, 1, \dots, n-1\}, \Sigma, \cdot_B, 0, \{n-1\})$ shown in Fig. 5. Notice that transitions on a and b in A are the same as in the NFA in Fig. 1 and perform the identity function in B . The roles of the transitions on c and d in B are the same as the roles of a and b in A . Therefore, it follows from Lemma 2 that for every subset S of $\{q_0, q_1, \dots, q_{m-1}\}$, there is a word u_S in $\{a, b\}^*$ such that $q_0 \cdot_A u_S = S$, and for every subset T of $\{0, 1, \dots, n-1\}$, there is a word v_T in $\{c, d\}^*$ such that $0 \cdot_B v_T = T$. Moreover, the words u_S satisfy the conditions (1), (2), (3) in Lemma 2.

To get an NFA N recognising KL from NFAs A and B , add the transitions $(q_{m-2}, a, 0)$, $(q_{m-1}, b, 0)$, and $(q_{m-1}, c, 0)$. The initial state of N is $\{q_0\}$ and its unique final state is $n-1$. Let $S \subseteq \{q_0, q_1, \dots, q_{m-1}\}$ and $T \subseteq \{0, 1, \dots, n-1\}$

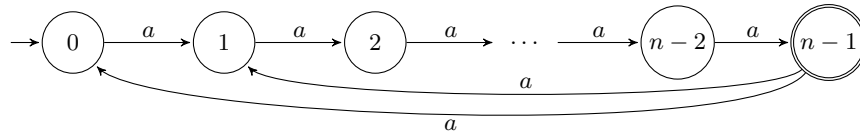


Fig. 4. A possible unary witness for star meeting the bound $(n-1)^2 + 2$.

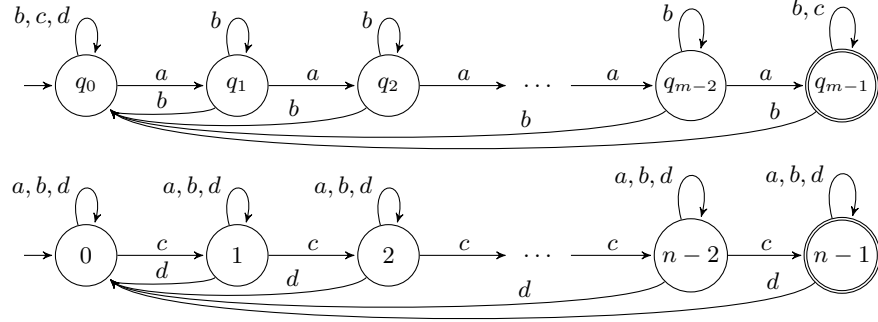


Fig. 5. Witness NFAs for concatenation meeting the upper bound $\frac{3}{4}2^{m+n}$.

be two subsets such that if $q_{m-1} \in S$ then $0 \in T$. The following transitions use the words $u_S \in \{a, b\}^*$ and $v_T \in \{c, d\}^*$ given by Lemma 2 to show that the set $S \cup T$ is reachable in the subset automaton $\mathcal{D}(N)$:

$$\begin{aligned} \{q_0\} &\xrightarrow{a^{m-1}} \{q_{m-1}, 0\} \xrightarrow{b} \{q_0, q_{m-1}, 0\} \xrightarrow{d} \{q_0, 0\}; \\ \{q_0, 0\} &\xrightarrow{v_T} \{q_0\} \cup T \xrightarrow{u_S} S \cup T \quad \text{if } S \neq \emptyset; \\ \{q_0, 0\} &\xrightarrow{a} \{q_1, 0\} \xrightarrow{d} \{0\} \xrightarrow{v_T} T; \end{aligned}$$

let us emphasise that by Lemma 2, while reading u_S , the final state q_{m-1} of A is not visited, except for the last step if $q_0 \notin S$ and $q_{m-1} \in S$ and when we must have $0 \in T$, and for the last two steps if $q_0 \in S$ and $q_{m-1} \in S$ when u_S ends with b that fixes the initial state 0 of B which must be in T . This proves the reachability of $\frac{3}{4}2^{m+n}$ states in the subset automaton $\mathcal{D}(N)$.

To prove distinguishability, notice that each singleton set $\{j\}$, $0 \leq j \leq n-1$ is co-reachable in N via a word in c^* . Next, $\{q_{m-1}\}$ is co-reachable by c^n , and each $\{q_i\}$, $0 \leq i \leq m-2$ is co-reachable via a word in c^na^* . By Lemma 1, all states of $\mathcal{D}(N)$ are pairwise distinguishable. \square

The witness in the previous proof is defined over a quaternary alphabet. Consider binary languages $K = (a+b)^*a(a+b)^{m-2}$ and $L = (a+b)^{n-1}$ recognised by an m -state and n -state NFA, respectively. Then $KL = (a+b)^*a(a+b)^{m+n-3}$, the minimal DFA for which has 2^{m+n-4} states. This gives the lower bound $(1/16)2^{m+n}$ in the binary case which is asymptotically the same as the upper bound for quaternary case. In the unary case the upper bound is $2^{O(\sqrt{(m+n)\ln(m+n)})}$. A lower bound $1 + F(n-1)$ is given by languages $K = \{\varepsilon\}$ and L equal to the witness for determinization in the unary case; here $F(n)$ is Landau's function given by $F(n) = \max\{\text{lcm}(x_1, \dots, x_k) \mid n = x_1 + \dots + x_k\}$, and with $F(n) \approx 2^{\sqrt{n \ln n}}$.

4 Boolean Operations

Here we consider NFA-to-DFA trade-off for four binary Boolean operations. First, we recall some notions. We call a state q of a DFA $A = (Q, \Sigma, \cdot, s, F)$ a *sink state* if $q \cdot a = q$ for every letter $a \in \Sigma$. The state q is called *dead* if reading every word from the state q results in a non-accepting state of A .

To get an automaton recognising union, intersection, difference, or symmetric difference of two languages we use the product construction as described below. Let $A = (Q_A, \Sigma, \cdot_A, s_A, F_A)$ and $B = (Q_B, \Sigma, \cdot_B, s_B, F_B)$ be two DFAs over an alphabet Σ . Let $\circ \in \{\cap, \cup, \setminus, \oplus\}$. Then the language $L(A) \circ L(B)$ is recognised by the product automaton $M_\circ = (Q_A \times Q_B, \Sigma, \cdot, (s_A, s_B), F_\circ)$ where $(p, q) \cdot a = (p \cdot_A a, q \cdot_B a)$ for all $p \in Q_A$, $q \in Q_B$, and $a \in \Sigma$, and

$$F_\circ = \begin{cases} F_A \times F_B, & \text{if } \circ = \cap; \\ (F_A \times Q_B) \cup (Q_A \times F_B), & \text{if } \circ = \cup; \\ F_A \times (Q_B \setminus F_B), & \text{if } \circ = \setminus; \\ (F_A \times (Q_B \setminus F_B)) \cup ((Q_A \setminus F_A) \times F_B), & \text{if } \circ = \oplus. \end{cases}$$

If the operation inputs are given by NFAs, we first apply the subset construction to get DFAs for those inputs. Then we construct the corresponding product automaton. Notice that every subset automaton has at least one rejecting sink state, namely, the empty set. The following lemma provides upper bounds for Boolean operations on DFAs considering the presence of the rejecting sink states.

Lemma 7. *Let K and L be languages over Σ accepted by DFAs with m and n states respectively. Assume that both DFAs have a rejecting sink state. Then $\text{sc}(K \cup L) \leq mn$, $\text{sc}(K \oplus L) \leq mn$, $\text{sc}(K \cap L) \leq mn - m - n + 2$, and $\text{sc}(K \setminus L) \leq mn - n + 1$.*

Proof. For each Boolean operation $\circ \in \{\cup, \cap, \setminus, \oplus\}$, the language $K \circ L$ is recognised by the product automaton M_\circ which has mn states. This gives the upper bounds for union and symmetric difference. Let d_A and d_B be the rejecting sink states of A and B , respectively. Then in the product automaton M_\cap recognizing $K \cap L$, the states (d_A, q) with $q \in Q_B$ and the states (p, d_B) with $p \in Q_A$ are dead and can be merged into one sink state. This gives the upper bound $(m-1)(n-1)+1 = mn - m - n + 2$. In the product automaton M_\setminus recognising $K \setminus L$, the states (d_A, p) with $p \in Q_B$ are dead, which gives the upper bound $(m-1)n+1 = mn - n + 1$. \square

Now we are ready to get tight upper bounds on NFA-to-DFA trade-off for Boolean operations.

Theorem 8. *Let K and L be languages over Σ recognised by an m -state and n -state NFA, respectively, where $m, n \geq 2$. Then $\text{sc}(K \cup L), \text{sc}(K \oplus L) \leq 2^{m+n}$, $\text{sc}(K \cap L) \leq 2^{m+n} - 2^m - 2^n + 2$, and $\text{sc}(K \setminus L) \leq 2^{m+n} - 2^n + 1$. All these bounds are tight if $|\Sigma| \geq 4$.*

Proof. Let A be an m -state NFA recognising K and B be an n -state NFA recognising L . Consider the corresponding subset automata $\mathcal{D}(A)$ and $\mathcal{D}(B)$ with 2^m and 2^n states, respectively. Both of them have at least one rejecting sink state, namely, the empty set. Then all upper bounds follow from Lemma 7.

For tightness, let K and L be the languages recognised by NFAs A and B from Fig. 6. Notice that transitions on a and b in A are the same as in the NFA in Fig. 1 and perform the identity function in B . The roles of the transitions on c and d in B are the same as the roles of a and b in A . Moreover c and d perform the identity function in A . It follows from Lemma 2 that for every $S \subseteq \{0, 1, \dots, m-1\}$, there is a word $u_S \in \{a, b\}^*$ such that $0 \cdot_A u_S = S$, and for every $T \subseteq \{0, 1, \dots, n-1\}$, there is a word $v_T \in \{c, d\}^*$ such that $0 \cdot_B v_T = T$.

Let $\circ \in \{\cup, \oplus, \cap, \setminus\}$. Construct the product automaton M_\circ from DFAs $\mathcal{D}(A)$ and $\mathcal{D}(B)$. The initial state of M_\circ is $(\{0\}, \{0\})$. Let $S \subseteq \{0, 1, \dots, m-1\}$ and $T \subseteq \{0, 1, \dots, n-1\}$. Then the state (S, T) is reachable in M_\circ from the initial state by the word $u_S v_T$. Hence each state of M_\circ is reachable.

To prove distinguishability first consider union. Then (S, T) is final in M_\cup if $m-1 \in S$ or $n-1 \in T$. Let (S, T) and (S', T') be two distinct states of M_\cup . Then $S \neq S'$ or $T \neq T'$. In the first case, without loss of generality let $s \in S \setminus S'$. Consider the word $a^{m-1-s}c^n$. Notice that

$$\begin{aligned} (S, T) &\xrightarrow{a^{m-1-s}c^n} (\{m-1\} \cup S_1, \emptyset) \text{ for some } S_1 \subseteq \{0, 1, \dots, m-1\}; \\ (S', T') &\xrightarrow{a^{m-1-s}c^n} (S'_1, \emptyset) \text{ where } m-1 \notin S'_1. \end{aligned}$$

It follows that $a^{m-1-s}c^n$ is accepted by M_\cup from (S, T) and rejected from (S', T') . The case of $T \neq T'$ is symmetric. We can prove distinguishability for symmetric difference in the exact same manner. We can also find the appropriate words that distinguish the desired number of states in the product automaton for intersection and difference. \square

Notice that the upper bounds in the previous theorem cannot be met by unary languages. The cases of binary and ternary alphabets remain open.

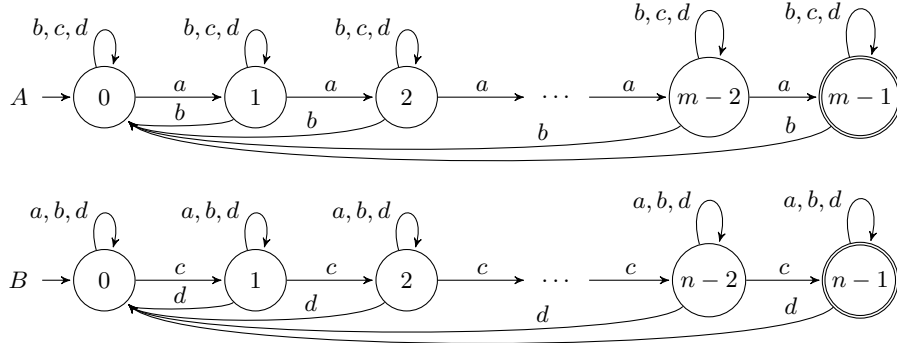


Fig. 6. Quaternary witnesses for Boolean operations.

5 Shuffle, Left and Right Quotient

Here we consider three more binary operations. In all three cases the upper bound constructions are similar to the case when the operation inputs are given by DFAs. Our lower bound for shuffle is the same as the upper bound on its state complexity. The lower bound for left quotient is greater by one than its state complexity. For right quotient, the NFA-to-DFA trade-off is 2^m , while its (nondeterministic) state complexity is m .

Theorem 9 (Shuffle). *Let K and L be languages over Σ recognised by an m -state and n -state NFA, respectively, where $m, n \geq 3$. Then $\text{sc}(K \sqcup L) \leq 2^{mn}$, and the bound is tight if $|\Sigma| \geq 4$.*

Proof. Let $A = (Q_A, \Sigma, s_A, \cdot_A, F_A)$ and $B = (Q_B, \Sigma, s_B, \cdot_B, F_B)$ be an m -state and n -state NFAs recognising the languages K and L , respectively. Then the language $K \sqcup L$ is recognised by mn -state NFA

$$N = (Q_A \times Q_B, \Sigma, \cdot, (s_A, s_B), F_A \times F_B)$$

where for each $(p, q) \in Q_A \times Q_B$ and $a \in \Sigma$,

$$(p, q) \cdot a = \{(p, q') \mid q' \in q \cdot_B a\} \cup \{(p', q) \mid p' \in p \cdot_A a\}.$$

It follows that $\text{sc}(K \sqcup L) \leq 2^{mn}$.

The m -state and n -state partial DFAs over $\{a, b, c, d, f\}$ meeting this upper bound have been described in [4, Proof of Theorem 1]. Notice in that the role of c and d in that proof is to reach the set $Q_A \times Q_B$ in the subset automaton. The same goal can be achieved if we replace c and d by the transitions on letter e defined as follows: $0 \cdot_A e = \{0, 1\}$, $i \cdot_A e = \{i + 1\}$, if $1 \leq i \leq m - 2$, and $0 \cdot_B e = \{0, 1\}$, $j \cdot_B e = \{j + 1\}$, if $1 \leq j \leq n - 2$. As a result we get a quaternary witness for shuffle. \square

In the unary case the upper bound for shuffle is again $2^{O(\sqrt{(m+n) \ln(m+n)})}$.

Theorem 10 (Left and Right Quotient). *Let K and L be languages over an alphabet Σ recognised by an m -state and n -state NFA, respectively, where $m, n \geq 2$. Then $\text{sc}(L^{-1}K), \text{sc}(KL^{-1}) \leq 2^m$, and the bounds are tight if $|\Sigma| \geq 2$.*

Proof. Let $A = (Q_A, \Sigma, s_A, \cdot_A, F_A)$ be an m -state NFA recognising K . The language $L^{-1}K$ is recognised by the m -state NFA N obtained from A by changing the set of initial states to $\{s_A \cdot_A w \mid w \in L\}$. The language KL^{-1} is recognised by the m -state NFA N obtained from A by changing the set of final states to $\{q \in Q_A \mid q \cdot_A w \in F_A \text{ for some } w \in L\}$. Hence $\text{sc}(L^{-1}K), \text{sc}(KL^{-1}) \leq 2^m$.

For tightness, notice that $\{\varepsilon\}^{-1}K = K\{\varepsilon\}^{-1} = K$. Therefore, the upper bound 2^m is met in both cases by $L = \{\varepsilon\}$ and K equal to the binary m -state witness NFA for determinization given by Lemma 2; for distinguishability, notice that each singleton set is co-reachable in this NFA. \square

The binary alphabet used in the theorem above is optimal since determinization in the unary case is in $2^{O(\sqrt{n \ln n})}$.

6 Conclusions

We investigated the NFA-to-DFA trade-off for several regular operations. Our results are summarised in Table 1. The table also displays the size of alphabet used to describe our witnesses. Whenever we used a binary alphabet, it was always optimal in the sense that the corresponding upper bounds cannot be met by any unary languages. The table also compares our results to the known results on the state complexity and the nondeterministic state complexity of all considered operations [3, 6, 10, 11, 15].

Table 2 shows the operational state complexity on languages represented by self-verifying and unambiguous finite automata from [8, 9]. The NFA-to-DFA trade-off for concatenation, shuffle, left and right quotient is up to one state almost the same as the complexity of these operations on unambiguous automata. The same holds for left quotient on self-verifying automata.

	NFA-to-DFA	$ \Sigma $	DFA	$ \Sigma $	NFA	$ \Sigma $
complementation	2^n	2	n	1	2^n	2
reversal	2^n	2	2^n	2	$n + 1$	2
star	2^n	2	$\frac{3}{4} \cdot 2^n$	2	$n + 1$	1
concatenation	$\frac{3}{4} 2^{m+n}$	4	$m 2^n - 2^{n-1}$	2	$m + n$	2
union	2^{m+n}	4	mn	2	$m + n + 1$	2
symmetric difference	2^{m+n}	4	mn	2	?	
intersection	$2^{m+n} - 2^m - 2^n + 2$	4	mn	2	mn	2
difference	$2^{m+n} - 2^n + 1$	4	mn	2	?	
shuffle	2^{mn}	4	$\leq 2^{mn}$	5	mn	2
left quotient	2^m	2	$2^m - 1$	2	$m + 1$	2
right quotient	2^m	2	m	1	m	1

Table 1. The NFA-to-DFA trade-off vs (nondeterministic) state complexity.

	SVFA	$ \Sigma $	UFA	$ \Sigma $
complementation	n	1	$n^{2-\epsilon} \leq \cdot \leq 2^{0.79n+\log n}$	1
reversal	$2n + 1$	2	n	1
star	$\frac{3}{4} \cdot 2^n$	$\frac{3}{4} \cdot 2^n + 1$	$\frac{3}{4} \cdot 2^n$	3
concatenation	$\Theta(3^{\frac{m}{3}} 2^n)$	$3^{\frac{m}{3}} + 2^n + 1$	$\frac{3}{4} \cdot 2^{m+n} - 1$	7
union	mn	2	$mn + m + n \leq \cdot \leq m + n 2^{0.79n+\log n}$	4
symmetric difference	mn	2	?	
intersection	mn	2	mn	2
difference	mn	2	?	
shuffle	?		$2^{mn} - 1$	5
left quotient	$2^m - 1$	$2^m + 1$	$2^m - 1$	2
right quotient	$3^{\frac{m}{3}}$	$3^{\frac{m}{3}} + 2$	$2^m - 1$	2

Table 2. Operational complexity for self-verifying and unambiguous automata.

Acknowledgement

We would like to kindly thank Michal Hospodár for his valuable notes and comments.

References

1. Birget, J.: Intersection and union of regular languages and state complexity. *Inform. Process. Lett.* **43**(4), 185–190 (1992). [https://doi.org/10.1016/0020-0190\(92\)90198-5](https://doi.org/10.1016/0020-0190(92)90198-5)
2. Birget, J.: The state complexity of $\overline{\Sigma^*L}$ and its connection with temporal logic. *Inform. Process. Lett.* **58**(4), 185–188 (1996). [https://doi.org/10.1016/0020-0190\(96\)00044-0](https://doi.org/10.1016/0020-0190(96)00044-0)
3. Brzozowski, J.A.: Quotient complexity of regular languages. *J. Autom. Lang. Comb.* **15**(1/2), 71–89 (2010). <https://doi.org/10.25596/jalc-2010-071>
4. Cămpăanu, 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). <https://doi.org/10.25596/jalc-2002-303>
5. Chrobak, M.: Finite automata and unary languages. *Theoret. Comput. Sci.* **47**(3), 149–158 (1986). [https://doi.org/10.1016/0304-3975\(86\)90142-8](https://doi.org/10.1016/0304-3975(86)90142-8)
6. Holzer, M., Kutrib, M.: Nondeterministic descriptive complexity of regular languages. *Internat. J. Found. Comput. Sci.* **14**(6), 1087–1102 (2003). <https://doi.org/10.1142/S0129054103002199>
7. Hospodár, M., Jirásková, G., Mlynárčik, P.: Descriptive complexity of the forever operator. *Internat. J. Found. Comput. Sci.* **30**(1), 115–134 (2019). <https://doi.org/10.1142/S0129054119400069>
8. Jirásek, J.S., Jirásková, G., Szabari, A.: Operations on self-verifying finite automata. In: Beklemishev, L.D., Musatov, D.V. (eds.) *CSR 2015. LNCS*, vol. 9139, pp. 231–261. Springer (2015). https://doi.org/10.1007/978-3-319-20297-6_16
9. Jirásek, J., Jr., Jirásková, G., Šebej, J.: Operations on unambiguous finite automata. *Internat. J. Found. Comput. Sci.* **29**(5), 861–876 (2018). <https://doi.org/10.1142/S012905411842008X>
10. Jirásková, G.: State complexity of some operations on binary regular languages. *Theoret. Comput. Sci.* **330**(2), 287–298 (2005). <https://doi.org/10.1016/j.tcs.2004.04.011>
11. Maslov, A.N.: Estimates of the number of states of finite automata. *Soviet Math. Doklady* **11**(5), 1373–1375 (1970)
12. Salomaa, A., Salomaa, K., Yu, S.: State complexity of combined operations. *Theoret. Comput. Sci.* **383**(2-3), 140–152 (2007). <https://doi.org/10.1016/j.tcs.2007.04.015>
13. Šebej, J.: Reversal of regular languages and state complexity. In: Pardubská, D. (ed.) *Proc. Conference on Theory and Practice of Information Technologies, ITAT 2010. CEUR Workshop Proceedings*, vol. 683, pp. 47–54. CEUR-WS.org (2010). <http://ceur-ws.org/Vol-683/paper8.pdf>
14. Yu, S.: Chapter 2: Regular languages. In: Rozenberg, G., Salomaa, A. (eds.) *Handbook of Formal Languages*, Vol. 1, pp. 41–110. Springer, Heidelberg (1997). https://doi.org/10.1007/978-3-642-59136-5_2
15. Yu, S., Zhuang, Q., Salomaa, K.: The state complexities of some basic operations on regular languages. *Theoret. Comput. Sci.* **125**(2), 315–328 (1994). [https://doi.org/10.1016/0304-3975\(92\)00011-F](https://doi.org/10.1016/0304-3975(92)00011-F)