State Complexity of Union and Intersection

Abstract. We investigate the state complexity and the nondeterministic state complexity of union and intersection of regular languages. We show that the entire range of complexities (nondeterministic complexities, respectively) of languages resulting from these operations, up to the known upper bounds, can be produced. We prove the results on the state complexity and the nondeterministic state complexity of union for a binary alphabet and the result on nondeterministic state complexity of intersection for a ternary alphabet.

1 Introduction

Regular languages and finite automata are among the oldest topics in computer science. Many of their properties have been extensively studied since the forties. In recent years, a renewed interest for researchers in automata theory can be observed; for a discussion, the reader may refer to [10, 23]. Some aspects of this area are now intensively and deeply investigated. One such aspect is the state complexity of regular languages and their operations.

The state complexity of a regular language is the number of states of its minimal deterministic finite automaton (DFA). The nondeterministic state complexity of a regular language is the number of states of a minimum state nondeterministic finite automaton (NFA) accepting the language. The state complexity (the nondeterministic state complexity) of an operation on regular languages represented by DFAs (NFAs, respectively) is the number of states that are sufficient and necessary in the worst case for a DFA (an NFA, respectively) to accept the language resulting from the operation.

Some early results on the state complexity of regular languages can be found in [13, 14, 16]. The state complexity of some operations on regular languages was investigated in [12, 1, 2]. In [15], binary regular languages reaching the upper bounds on the union, concatenation and the star operation were presented. The first systematic study of the complexity of regular language operations was published by Yu, Zhuang, and Salomaa [21]. The state complexity of finite language operations and unary language operations was investigated in [3, 17]. The nondeterministic state complexity of regular language operations was studied by Holzer and Kutrib in [8]. Further results on this topic are presented in [5, 4, 6, 11, 19] and state-of-the-art surveys for DFAs can be found in [24, 25].

In this paper, we investigate the state complexity and the nondeterministic state complexity of union and intersection of regular languages. We treat the question of how many states may the minimal DFA accepting the union or intersection of an m-state DFA language and an n-state DFA language have. We deal with the same question for nondeterministic state complexity of these two operations. Our results are as follows. The state complexity of the union or intersection of an m-state

DFA language and an n-state DFA language may be arbitrary between 1 and mn. The nondeterministic state complexity of the union of an m-state DFA language and an n-state DFA language may be arbitrary between 1 and $m + n + 1$. The nondeterministic state complexity of the intersection of an m-state DFA language and an n-state DFA language may be arbitrary between 1 and mn. We prove the results on the state complexity and the nondeterministic state complexity of union for a binary alphabet and the result on the nondeterministic state complexity of intersection for a ternary alphabet. To prove the results in the nondeterministic case we use a fooling-set lower-bound technique known from communication complexity theory [9], cf. also [1, 2, 7].

The paper consists of six sections, including this introduction. The next section contains basic definitions and notations used throughout the paper. In Section 3, we present our results on the state complexity of union and intersection. Section 4 deals with the nondeterministic state complexity of union of regular languages. Section 5 is devoted to the nondeterministic state complexity of intersection of regular languages. The last section contains concluding remarks and open problems.

2 Preliminaries

In this section, we recall some basic definitions and notations. For further details, the reader may refer to [20, 22].

Let Σ be an alphabet and Σ^* the set of all strings over the alphabet Σ including the empty string ε. The length of a string w is denoted by $|w|$. The power-set of a finite set A is denoted by 2^A.

A *deterministic finite automaton* (DFA) is a 5-tuple $M = (Q, \Sigma, \delta, q_0, F)$, where Q is a finite set of states, Σ is a finite input alphabet, $\delta : Q \times \Sigma \to Q$ is the transition function, $q_0 \in Q$ is the initial state, and $F \subseteq Q$ is the set of accepting states. In this paper, all DFAs are assumed to be complete, i.e., the next state $\delta(q, a)$ is defined for any state q in Q and any symbol a in Σ. The transition function δ can be naturally extended to a function from $Q \times \Sigma^*$ to Q. A string w in Σ^* is accepted by the DFA M if the state $\delta(q_0, w)$ is an accepting state of the DFA M.

A *nondeterministic finite automaton* (NFA) is a 5-tuple $M = (Q, \Sigma, \delta, q_0, F)$, where Q, Σ, q_0, and F are defined as for a DFA, and $\delta : Q \times \Sigma \to 2^Q$ is the transition function which can be extended to the domain $Q \times \Sigma^*$. A string w in Σ^* is accepted by the NFA M if the set $\delta(q_0, w)$ contains an accepting state of the NFA M.

The *language accepted by* a finite automaton M, denoted $L(M)$, is the set of all strings accepted by the automaton M. Two automata are said to be *equivalent* if they accept the same language.

A DFA (an NFA) M is called *minimal* if all DFAs (all NFAs, respectively) that are equivalent to M have at least as many states as M. By a well-known result, each regular language has a unique minimal DFA, up to isomorphism. However, the same result does not hold for minimal NFAs. It is also known [18] that a DFA $M = (Q, \Sigma, \delta, q_0, F)$ is minimal if (i) all its states are reachable from the initial state q_0 and (ii) no two of its states are equivalent (two states p and q are said to be *equivalent* if for all $w \in \Sigma^*$, $\delta(p, w) \in F$ iff $\delta(q, w) \in F$).

3 Union and Intersection and Their State Complexity

We start our investigations with the state complexity of the union and intersection of regular languages. It is known [15, 21] that the union of an m-state DFA language and an n-state DFA language can be accepted by an mn-state DFA. This upper bound can be reached by the union of two binary languages [15]. The same result for intersection follows from de Morgan's law and the fact that the state complexity of a regular language equals the state complexity of its complement.

In this section, we show that for any integers m, n, α such that $m \geqslant 2, n \geqslant 2$, and $1 \leqslant \alpha \leqslant mn$, there exist an m-state DFA language and an n-state DFA language such that the minimal DFA for the union of these languages has exactly α states. The same result for intersection then follows immediately. In the case of $m = 1$, the language accepted by a 1-state DFA is either empty or equals Σ^*, and so we get the following result.

Proposition 1. *For any positive integer n, the state complexity of the union of a 1-state DFA language and an n-state DFA language may be either 1 or n.*

In the following two lemmata, we assume $m \leqslant n$. The first lemma shows that all values of α between 1 and m can be reached by the union of an m-state DFA language and an n-state DFA language. The second lemma shows the same result for the values between $m + 1$ and $m + n - 2$. To prove the results we use a unary alphabet in Lemma 1 and a binary alphabet in Lemma 2.

Lemma 1. *For any integers m, n, α such that $1 \leqslant \alpha \leqslant m \leqslant n$, there exist a minimal DFA A of m states and a minimal DFA B of n states such that the minimal DFA for the language $L(A) \cup L(B)$ has α states.*

Proof. Let m, n, and α be arbitrary but fixed integers with $1 \leqslant \alpha \leqslant m \leqslant n$. Let $\Sigma = \{a\}$. Let $A = (\{q_0, q_1, \ldots, q_{m-1}\}, \Sigma, \delta_A, q_0, \{q_{m-1}\})$, where $\delta(q_i, a) = q_{i+1}$ for $i = 0, 1, \ldots, n - 2$, and $\delta(q_{m-1}, a) = q_{m-1}$. The DFA A accepts the language $\{a^i \mid i \geqslant m - 1\}$ and is minimal since it does not contain equivalent states. In the case of $\alpha = m = n$, we set $B = A$ and then the lemma follows. Otherwise, let $B = (\{p_0, p_1, \ldots, p_{n-1}\}, \Sigma, \delta_B, p_0, \{p_{\alpha-1}, p_\alpha, \ldots, p_{n-2}\})$, where $\delta_B(p_i, a) = p_{i+1}$ for $i = 0, 1, \ldots, n - 2$, and $\delta_B(p_{n-1}, a) = p_{n-1}$. The DFA B is the minimal DFA for the language $\{a^i \mid \alpha - 1 \leqslant i \leqslant n - 2\}$. The union of the languages accepted by the DFA A and B is the language $L(A) \cup L(B) = \{a^i \mid i \geqslant \alpha - 1\}$. The minimal DFA for this language has α states. □

Lemma 2. *For any integers m, n, α such that $2 \leqslant m \leqslant n$ and $m+1 \leqslant \alpha \leqslant m+n-2$, there exist a minimal DFA A of m states and a minimal DFA B of n states such that the minimal DFA for the language $L(A) \cup L(B)$ has α states.*

Proof. Let m, n, and α be arbitrary but fixed integers such that $2 \leqslant m \leqslant n$ and $m + 1 \leqslant \alpha \leqslant m + n - 2$. Then α can be expressed as $\alpha = m + k$, where $1 \leqslant k \leqslant n - 2$. Let $\Sigma = \{a, b\}$.

Let B be the minimal n-state DFA over Σ for the language $\{b^i \mid i \geqslant k\} \cup \{a^{n-k-2}\}$. To define the DFA A we will consider two cases:

(i) $m \leqslant n-k$. Let A be the minimal m-state DFA over the alphabet Σ for the language $\{a^i \mid i \geqslant m-2\}$. Since $m \leqslant n-k$, the union of the languages accepted by the DFA A and B is the languauge $L(A) \cup L(B) = \{a^i \mid i \geqslant m-2\} \cup \{b^i \mid i \geqslant k\}$. The minimal DFA for this language has $m+k$ states.

(ii) $m > n-k$. Let A be the minimal m-state DFA over the alphabet Σ for the language $\{a^{m-2}\}$. Since $m > n-k$, the union of the languages accepted by the DFA A and B is the languauge $L(A) \cup L(B) = \{a^{n-k-2}, a^{m-2}\} \cup \{b^i \mid i \geqslant k\}$. The minimal DFA for this language has $m+k$ states. □

The next lemma shows that all the values between $m+n-1$ and mn can be reached by the union of an m-state DFA language and an n-state DFA language over a ternary alphabet.

Lemma 3. *For any integers m, n, α with $m \geqslant 2, n \geqslant 2$, and $m+n-1 \leqslant \alpha \leqslant mn$, there exist a minimal DFA A of m states and a minimal DFA B of n states such that the minimal DFA for the language $L(A) \cup L(B)$ has α states.*

Proof. Let m, n, and α be arbitrary but fixed integers such that $m \geqslant 2, n \geqslant 2$, and $m+n-1 \leqslant \alpha \leqslant mn$. Then α can be expressed as $\alpha = m + s(n-1) + t$, where $1 \leqslant s \leqslant m-1$ and $0 \leqslant t \leqslant n-1$. Let $\Sigma = \{a, b, c\}$.

Define an m-state DFA $A = (Q_A, \Sigma, \delta_A, q_0, F_A)$, where $Q_A = \{q_0, q_1, \ldots, q_{m-1}\}$, $F_A = \{q_{m-1}\}$, and for any $i \in \{0, 1, \ldots, m-1\}$ and any $X \in \Sigma$,

$$\delta_A(q_i, X) = \begin{cases} q_{(i+1) \bmod m}, & \text{if } X = a, \\ q_i, & \text{if } i < s \text{ and } X = b, \\ q_0, & \text{if } i \geqslant s \text{ and } X = b, \\ q_i, & \text{if } i = s \text{ and } X = c, \\ q_0, & \text{if } i \neq s \text{ and } X = c. \end{cases}$$

Define an n-state DFA $B = (Q_B, \Sigma, \delta_B, p_0, F_B)$, where $Q_B = \{p_0, p_1, \ldots, p_{n-1}\}$, $F_B = \{p_{n-1}\}$, and for any $i \in \{0, 1, \ldots, n-1\}$ and any $X \in \Sigma$,

$$\delta_B(p_i, X) = \begin{cases} p_0, & \text{if } X = a, \\ p_{(i+1) \bmod n}, & \text{if } X = b, \\ p_{i+1}, & \text{if } i < t \text{ and } X = c, \\ p_0, & \text{if } i \geqslant t \text{ and } X = c. \end{cases}$$

The DFA A and B are shown in Fig. 1. Both automata are minimal since the string a^{m-1-j} distinguishes states q_i and q_j with $i < j$, and the string b^{n-1-j} distinguishes states p_i and p_j with $i < j$.

Let $C = (Q_A \times Q_B, \Sigma, \delta, [q_0, p_0], F)$ be the cross-product of automata A and B accepting the language $L(A) \cup L(B)$. We are going to prove that the DFA C has exactly α reachable states no two of which are equivalent. Let $\mathcal{R}$ be the following set of states of the DFA C

$$\mathcal{R} = \{[q_i, p_0] \mid 0 \leqslant i < m\} \cup \{[q_i, p_j] \mid 0 \leqslant i < s, 1 \leqslant j < n\} \cup \{[q_s, p_j] \mid 1 \leqslant j \leqslant t\}.$$

There are $m + s(n-1) + t$ states in the set $\mathcal{R}$. Any state in $\mathcal{R}$ is reachable in the DFA C because we have:

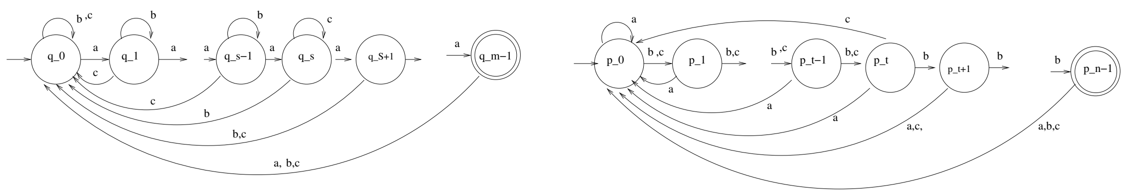

Fig. 1. The deterministic finite automata A and B.

$[q_i, p_0] = \delta([q_0, p_0], a^i)$ for $i = 0, 1, \ldots, m-1$,
$[q_i, p_j] = \delta([q_0, p_0], a^i b^j)$ for $i = 0, 1, \ldots, s-1$ and $j = 1, 2, \ldots, n-1$, and
$[q_s, p_j] = \delta([q_0, p_0], a^s c^j)$ for $j = 1, 2, \ldots, t$.

To prove that no other state is reachable in the DFA C note that the initial state $[q_0, p_0]$ of the DFA C is in $\mathcal{R}$ and for any state $[q, p]$ in $\mathcal{R}$ and any symbol X in Σ the state $\delta([q, p], X)$ is in $\mathcal{R}$ as well.

To prove that no two different states in $\mathcal{R}$ are equivalent let $[q_i, p_j]$ and $[q_k, p_l]$ be two different states in $\mathcal{R}$. Then, either $i \neq k$ or $j \neq l$. In the first case, these states are distinguished by the string a^{m-1-i}, and in the second case by the string b^{n-1-j}. □

To prove the result in the lemma above we used a three-letter alphabet. The next lemma shows that the same result holds for a binary alphabet as well.

Lemma 4. *For any integers m, n, α with $m \geqslant 2$, $n \geqslant 2$, and $m + n - 1 \leqslant \alpha \leqslant mn$, there exist a minimal binary DFA A of m states and a minimal binary DFA B of n states such that the minimal DFA for the language $L(A) \cup L(B)$ has α states.*

Proof. Let m, n, and α be arbitrary but fixed integers such that $m \geqslant 2$, $n \geqslant 2$, and $m + n - 1 \leqslant \alpha \leqslant mn$. Let $\Sigma = \{a, b\}$.

First, let α can be expressed as $\alpha = rm + sn - rs$ for some integers r and s such that $1 \leqslant r \leqslant n$ and $1 \leqslant s \leqslant m$.

Define an m-state DFA $A_{rs} = (\{q_0, q_1, \ldots, q_{m-1}\}, \Sigma, \delta_A^{(rs)}, q_0, \{q_{m-1}\})$, where for any $i \in \{0, 1, \ldots, m-1\}$ and any $X \in \Sigma$,

$$\delta_A^{(rs)}(q_i, X) = \begin{cases} q_{(i+1) \bmod m}, & \text{if } X = a, \\ q_i, & \text{if } i < s \text{ and } X = b, \\ q_0, & \text{if } i \geqslant s \text{ and } X = b. \end{cases}$$

Define an n-state DFA $B_{rs} = (\{p_0, p_1, \ldots, p_{n-1}\}, \Sigma, \delta_B^{(rs)}, p_0, \{p_{n-1}\})$, where for any $i \in \{0, 1, \ldots, n-1\}$ and any $X \in \Sigma$,

$$\delta_B^{(rs)}(p_i, X) = \begin{cases} p_{(i+1) \bmod n}, & \text{if } X = b, \\ p_i, & \text{if } i < r \text{ and } X = a, \\ p_0, & \text{if } i \geqslant r \text{ and } X = a. \end{cases}$$

Let $C_{rs} = (Q, \Sigma, \delta^{(rs)}, [q_0, p_0], F)$ be the cross-product of automata A_{rs} and B_{rs} accepting the language $L(A) \cup L(B)$. We are going to prove that the DFA C_{rs} has exactly α reachable states no two of which are equivalent. Let $\mathcal{R}_{rs}$ be the following set of states: $\mathcal{R}_{rs} = \{[q_i, p_j] \mid 0 \leqslant i \leqslant s-1 \text{ or } 0 \leqslant j \leqslant r-1\}$. There are $rm+sn-rs$ states in the set $\mathcal{R}_{rs}$. Any state in $\mathcal{R}_{rs}$ is reachable in the DFA C_{rs} because we have $[q_i, p_j] = \delta([q_0, p_0], a^i b^j)$ if $i \leqslant s-1$, and $[q_i, p_j] = \delta([q_0, p_0], b^j a^i)$ if $j \leqslant r-1$. To prove that no other state is reachable in the DFA C_{rs} note that the initial state $[q_0, p_0]$ of the DFA C_{rs} is in $\mathcal{R}_{rs}$ and for any state $[q, p]$ in $\mathcal{R}_{rs}$ and any symbol X in Σ the state $\delta([q, p], X)$ is in $\mathcal{R}_{rs}$ as well.

To prove that no two different states in $\mathcal{R}_{rs}$ are equivalent let $[q_i, p_j]$ and $[q_k, p_l]$ be two different states in $\mathcal{R}_{rs}$. We will consider four cases:

(i) $r < n$ and $s < m$. The string a^{m-1-i} distinguishes states $[q_i, p_j]$ and $[q_k, p_l]$ if $i \neq k$ and the string b^{n-1-j} distinguishes these states if $j \neq l$.

(ii) $r < n$ and $s = m$. The string ab^{n-1-i} distinguishes states $[q_{m-1}, p_i]$ and $[q_{m-1}, p_j]$ with $i \neq j$. The other pairs of states can be distigiushed as in case (i).

(iii) $r = n$ and $s < m$. The string ba^{m-1-i} distinguishes states $[q_i, p_{n-1}]$ and $[q_j, p_{n-1}]$ with $i \neq j$. The other pairs of states can be distigiushed as in case (i).

(iv) $r = n$ and $s = m$. The string ab^{n-1-i} distinguishes states $[q_{m-1}, p_i]$ and $[q_{m-1}, p_j]$ with $i \neq j$. The string ba^{m-1-i} distinguishes states $[q_i, p_{n-1}]$ and $[q_j, p_{n-1}]$ with $i \neq j$. The string a distinguishes states $[q_{m-1}, p_i]$ and $[q_j, p_{n-1}]$. The other pairs of states can be distigiushed as in case (i).

Now, let r and s be an arbitrary integers such that $r < n$ and $s < m$. Let t be an integer such that $1 \leqslant t \leqslant \min\{m-s, n-r\} - 1$. We are going to define binary automata A and B such that the minimal DFA for the language $L(A) \cup L(B)$ has $rm + sn - rs + t$ states.

Define an m-state DFA $A = (\{q_0, q_1, \ldots, q_{m-1}\}, \Sigma, \delta_A, q_0, \{q_{m-1}\})$, where $\delta_A(q_{s+i}, b) = q_{s+i-1}$, if $i \leqslant t$ and i is odd, and $\delta_A(q_i, X) = \delta_A^{(rs)}(q_i, X)$, otherwise.

Define an n-state DFA $B = (\{p_0, p_1, \ldots, p_{n-1}\}, \Sigma, \delta_B, p_0, \{p_{n-1}\})$, where $\delta_B(p_{r+i}, a) = p_{r+i-1}$, if $i \leqslant t$ and i is even, and $\delta_B(p_i, X) = \delta_B^{(rs)}(p_i, X)$, otherwise. Then, the set of reachable and pairwise inequivalent states of the cross-product automaton for the language $L(A) \cup L(B)$ is $\mathcal{R}_{rs} \cup \{[q_s, p_{r+j}] \mid j \leqslant t \text{ and } j \text{ is even}\} \cup \{[q_{s+i}, p_r] \mid i \leqslant t \text{ and } i \text{ is odd}\}$.

Finally, note that if $t = \min\{m-s, n-r\}$, say $t = m-s$, then $rm+sn-rs+t = (r+1)m + sn - (r+1)s$, i.e., $rm + sn - rs + t$ is the size of the minimal DFA for the language $L(A_{r+1s}) \cup L(B_{r+1s})$. This completes our proof. $\square$

As a corollary of the four lemmata above, we get the following result.

Theorem 1. *For any integers m, n, α such that $m \geqslant 2$, $n \geqslant 2$, and $1 \leqslant \alpha \leqslant mn$, there exist a minimal binary DFA A of m states and a minimal binary DFA B of n states such that the minimal DFA for the language $L(A) \cup L(B)$ has α states.* $\square$

The same result for intersection follows from de Morgan's law and the fact that the state complexity of a regular language equals the state complexity of its complement.

Theorem 2. *For any integers m, n, α such that $m \geqslant 2$, $n \geqslant 2$, and $1 \leqslant \alpha \leqslant mn$, there exist a minimal binary DFA A of m states and a minimal binary DFA B of n states such that the minimal DFA for the language $L(A) \cap L(B)$ has α states.* $\square$

4 Union and Its Nondeterministic State Complexity

We now turn our attention to the nondeterministic state complexity of the union of regular languages. It is known [8] that the union of an m-state NFA language and an n-state NFA language can be accepted by an $(m+n+1)$-state NFA and this upper bound can be reached by the union of two binary regular languages.

In this section, we show that the nondeterministic complexity of the union of an m-state NFA language and an n-state NFA language may be arbitrary between 1 and $m+n+1$ except for the case of $m=1$ and $n=1$. To prove the result we use a fooling-set lower-bound technique known from communication complexity theory [9], cf. also [1, 2, 7]. After defining a fooling set, we give a lemma from [1] describing the lower-bound technique.

Definition 1. *A set of pairs of strings $\{(x_i, y_i) \mid i = 1, 2, \ldots, n\}$ is said to be a fooling set for a regular language L if for any i and j in $\{1, 2, \ldots, n\}$,*
(1) the string $x_i y_i$ is in the language L, and
(2) if $i \neq j$, then at least one of the strings $x_i y_j$ and $x_j y_i$ is not in the language L.

Lemma 5 (Birget [1]). *Let a set of pairs $\{(x_i, y_i) \mid i = 1, 2, \ldots, n\}$ be a fooling set for a regular language L. Then any NFA for the language L needs at least n states.*

The following lemma shows that all the values up to the minimum of m and n can be reached by the union of an m-state and an n-state NFA languages.

Lemma 6. *For any integers m, n, α such that $1 \leqslant \alpha \leqslant m \leqslant n$, there exist a minimal NFA A of m states and a minimal NFA B of n states such that minimal NFAs for the language $L(A) \cup L(B)$ have α states.*

Proof. Let m, n, and α be arbitrary but fixed integers such that $1 \leqslant \alpha \leqslant m \leqslant n$. Let $\Sigma = \{a\}$.

Let $A = (\{q_0, q_1, \ldots, q_{m-1}\}, \Sigma, \delta_A, q_0, \{q_{m-1}\})$, where $\delta_A(q_i, a) = \{q_{i+1}\}$ if $0 \leqslant i \leqslant m-2$, and $\delta_A(q_{m-1}, a) = \{q_{m-1}\}$, be an m-state NFA accepting the language $\{a^i \mid i \geqslant m-1\}$. To prove that the NFA A is minimal consider a set of pairs $\{(a^{i-1}, a^{m-i}) \mid i = 1, 2, \ldots, m\}$. It is a fooling set for the language $L(A)$ because for any i and j in $\{1, 2, \ldots, m\}$, (1) $a^{i-1}a^{m-i} = a^{m-1}$ and the string a^{m-1} is in the language $L(A)$, and (2) if $i < j$, then $a^{i-1}a^{m-j} = a^{m-1-(j-i)}$ and the string $a^{m-1-(j-i)}$ is not in the language $L(A)$ since $m-1-(j-i) < m-1$. By Lemma 5, any NFA for the language $L(A)$ needs at least m states, and so the NFA A is minimal.

Let $B = (\{p_0, p_1, \ldots, p_{n-1}\}, \Sigma, \delta_B, p_0, \{p_{\alpha-1}, p_\alpha, \ldots, p_{n-1}\})$, where $\delta_B(p_i, a) = \{p_{i+1}\}$ if $0 \leqslant i \leqslant n-2$, and $\delta_B(p_{n-1}, a) = \emptyset$, be an n-state NFA accepting the language $\{a^i \mid \alpha - 1 \leqslant i \leqslant n-1\}$. To prove that the NFA B is minimal consider a set of pairs $\{(a^{i-1}, a^{n-i}) \mid i = 1, 2, \ldots, n\}$. It is a fooling set for the language $L(B)$ because for any i and j in $\{1, 2, \ldots, n\}$, (1) $a^{i-1}a^{n-i} = a^{n-1}$ and the string a^{n-1} is in the language $L(B)$, and (2) if $i < j$, then $a^{j-1}a^{n-i} = a^{n-1+j-i}$ and the string $a^{n-1+j-i}$ is not in the language $L(B)$ since $n-1+j-i > n-1$. By Lemma 5, any NFA for the language $L(B)$ needs at least n states, and so the NFA B is minimal.

The union of the languages accepted by the NFA A and B is the language $L(A) \cup L(B) = \{a^i \mid i \geqslant \alpha - 1\}$. As shown in the case of the DFA A, minimal NFAs for this language have α states. □

The next two lemmata deal with the cases of $m = 1$ and $m = 2$. Note that the nondeterministic state complexity of two 1-state NFAs may be 1 or 3.

Lemma 7. *For any integers n, α such that $n \geqslant 2$ and $2 \leqslant \alpha \leqslant n+1$, there exist a 1-state NFA A and a minimal n-state NFA B such that minimal NFAs for the language $L(A) \cup L(B)$ have α states.*

Proof. Let $n \geqslant 2$ and let A be a 1-state NFA for the language a^*. If $\alpha = 2$, let B be a minimal n-state NFA for the language $\{a^{n-2}\} \cup \{ba^i \mid i \geqslant 0\}$. Then, $L(A) \cup L(B) = a^* \cup ba^*$, minimal NFAs for which have 2 states. If $3 \leqslant \alpha \leqslant n+1$, let B be a minimal n-state NFA for the language $\{b^{\alpha-2}\} \cup \{a^{n-\alpha+2}\}$. Then, $L(A) \cup L(B) = a^* \cup \{b^{\alpha-2}\}$. Minimal NFAs for this language have α states. □

Lemma 8. *For any integers n, α such that $n \geqslant 2$ and $3 \leqslant \alpha \leqslant n+1$, there exist a minimal 2-state NFA A and a minimal n-state NFA B such that minimal NFAs for the language $L(A) \cup L(B)$ have α states.*

Proof. Let $n \geqslant 2$ and $3 \leqslant \alpha \leqslant n+1$. Let A be a minimal 2-state NFA for the language a^+. Let B be a minimal n-state NFA for the language $\{b^{\alpha-2}\} \cup \{a^{n-\alpha-2}\}$. Then, $L(A) \cup L(B) = a^+ \cup \{b^{\alpha-2}\}$. Minimal NFAs for this language have α states. □

In the next lemma, we assume m and n to be at least 3 and show that all values between $m+1$ and $m+n-2$ can be reached by the union of appropriate NFAs.

Lemma 9. *For any integers m, n, α such that $3 \leqslant m \leqslant n$ and $m+1 \leqslant \alpha \leqslant m+n-2$, there exist a minimal NFA A of m states and a minimal NFA B of n states such that minimal NFAs for the language $L(A) \cup L(B)$ have α states.*

Proof. Let $3 \leqslant m \leqslant n$ and $m+1 \leqslant \alpha \leqslant m+n-2$. Then α can be expressed as $\alpha = m+k$, where $1 \leqslant k \leqslant n-2$. Let $\Sigma = \{a, b\}$. Let B be an n-state NFA accepting the language $\{a^{n-k-1}, b^{k+1}\}$. The NFA B is minimal since the set of pairs $\{(a^{i-1}, a^{n-k-i}) \mid i = 1, 2, \ldots, n-k\} \cup \{(b^i, b^{k+1-i}) \mid i = 1, 2, \ldots, k\}$ is a fooling set for the language $L(B)$. To define the DFA A we will consider two cases:

(i) $m \leqslant n-k$. Let A be a minimal m-state NFA for the language $\{a^i \mid i \geqslant m-1\}$. The union of the languages accepted by the NFA A and B is the language $\{a^i \mid i \geqslant m-1\} \cup \{b^{k+1}\}$ which is accepted by an $(m+k)$-state NFA C shown in Fig. 2 (left). The NFA C is minimal since the set $\{(a^{i-1}, a^{m-i}) \mid i = 1, 2, \ldots, m\} \cup \{(b^i, b^{k+1-i}) \mid i = 1, 2, \ldots, k\}$ is a fooling set for the language $L(A) \cup L(B)$.

(ii) $m > n-k$. Let A be a minimal m-state NFA for the language $\{a^{m-1}\}$. The union of languages accepted by the NFAs A and B is the language $\{a^{n-k-1}, a^{m-1}, b^{k+1}\}$ which is accepted by an $(m+k)$-state NFA C' shown in Fig. 2 (right). The NFA C' is minimal since the set of pairs of strings $\{(a^{i-1}, a^{m-i}) \mid i = 1, 2, \ldots, m\} \cup \{(b^i, b^{k+1-i}) \mid i = 1, 2, \ldots, k\}$ is a fooling set for the language $L(A) \cup L(B)$. □

The next two lemmata deal with the cases of $\alpha = m+n-1$ and $\alpha = m+n$.

Lemma 10. *For any integers m, n such that $m \geqslant 2$ and $n \geqslant 2$, there exist a minimal NFA A of m states and a minimal NFA B of n states such that minimal NFAs for the language $L(A) \cup L(B)$ have $m+n-1$ states.*

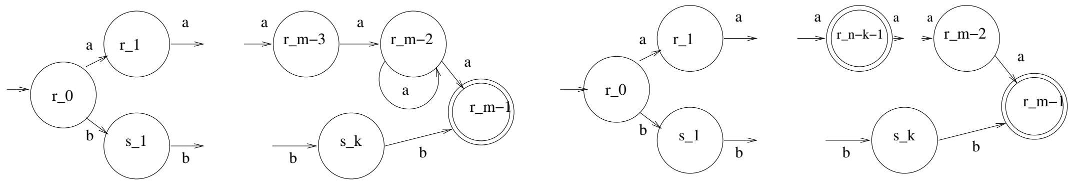

Fig. 2. The nondeterministic finite automata C and C'.

Proof. Let $m \geqslant 2$, $n \geqslant 2$, and let $\Sigma = \{a, b\}$. Let A be an m-state NFA shown in Fig. 3 (left) and let B be an n-state NFA shown in Fig. 3 (right). Both automata are minimal since the set of pairs of strings $\{(a^{i-1}, a^{m-i}) \mid i = 1, 2, \ldots, m\}$ is a fooling set for the language $L(A)$ and the set $\{(b^{i-1}, b^{n-i}) \mid i = 1, 2, \ldots, n\}$ is a fooling set for the language $L(B)$. The language $L(A) \cup L(B)$ is accepted by an $(m+n-1)$-state

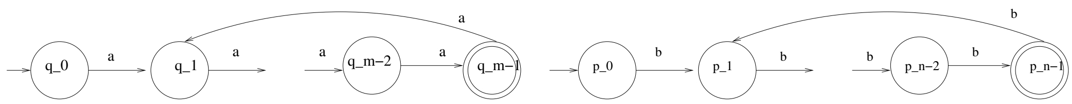

Fig. 3. The nondeterministic finite automata A and B.

NFA. Since the set $\{(a^{i-1}a^{m-i} \mid i = 1, 2, \ldots, m-1\} \cup \{(a^{m-1}, a^{m-1})\} \cup \{(b^i, b^{n-1-i}) \mid i = 1, 2, \ldots, n-1\}$ is a fooling set for the language $L(A) \cup L(B)$, minimal NFAs for this language have $m + n - 1$ states. □

Lemma 11. *For any positive integers m, n such that $n \geqslant 2$, there exist a minimal NFA A of m states and a minimal NFA B of n states such that minimal NFAs for the language $L(A) \cup L(B)$ have $m + n$ states.*

Proof. Let $m \geqslant 1$, $n \geqslant 2$, and let $\Sigma = \{a, b\}$. Let A be a minimal m-state NFA for the language $\{a^m\}^*$ and let B be a minimal n-state NFA for the language $\{b^{n-1}\}$. Then the union of the languages accepted by the NFA A and B is accepted by an $(m+n)$-state NFA. Since the set of pairs $\{(a^i, a^{m-1}) \mid i = 1, 2, \ldots, m-1\} \cup \{(a^m, a^m)\} \cup \{(b^{i-1}, b^{n-i}) \mid i = 1, 2, \ldots, n\}$ is a fooling set for the language $L(A) \cup L(B)$, minimal NFAs for this language have $m + n$ states. □

The following result is proved in [8].

Lemma 12. *For any positive integers m, n, minimal NFAs for the union of languages $\{a^m\}^*$ and $\{b^n\}^*$ have $m + n + 1$ states.*

As a corollary of the lemmata above, we get the following result.

Theorem 3. *For any positive integers m, n, α such that $m \geqslant 2$ or $n \geqslant 2$, and $1 \leqslant \alpha \leqslant m + n + 1$, there exist a minimal binary NFA A of m states and a minimal binary NFA B of n states such that minimal NFAs for the language $L(A) \cup L(B)$ have α states.* □

5 Intersection and Its Nondeterministic State Complexity

In this section, we study the nondeterministic state complexity of the intersection of regular languages. The intersection of an m-state NFA language and an n-state NFA language can be accepted by an mn-state NFA and this upper bound is known to be tight for a binary alphabet [8]. We show that the nondeterministic state complexity of the intersection of an m-state NFA language and an n-state NFA language may be arbitrary between 1 and mn. We prove the result for a ternary alphabet.

Lemma 13. *For any integers m, n, α such that $1 \leqslant \alpha \leqslant m \leqslant n$, there exist a minimal NFA A of m states and a minimal NFA B of n states such that minimal NFAs for the language $L(A) \cap L(B)$ have α states.*

Proof. Let $1 \leqslant \alpha \leqslant m \leqslant n$ and let A and B be m-state and n-state NFAs accepting languages $\{a, b\}^{\alpha-1} \cup \{a^{m-1}\}$ and $\{a, b\}^{\alpha-1} \cup \{b^{n-1}\}$, respectively. Both NFAs are minimal since the sets of pairs of strings $\{(a^{i-1}, a^{m-i}) \mid i = 1, 2, \dots, m\}$ and $\{(b^{i-1}, b^{n-i}) \mid i = 1, 2, \dots, n\}$ are fooling sets for the languages $L(A)$ and $L(B)$, respectively. The intersection of these languages is the language $\{a, b\}^{\alpha-1}$. Minimal NFAs for this language have α states. $\square$

Lemma 14. *For any positive integers m, n, α such that $m \leqslant n$ and $m \leqslant \alpha \leqslant mn$, there exist a minimal NFA A of m states and a minimal NFA B of n states such that minimal NFAs for the language $L(A) \cap L(B)$ have α states.*

Proof. Let m, n, and α be arbitrary but fixed positive integers such that $m \leqslant n$ and $m \leqslant \alpha \leqslant mn$. Then α can be expressed as $\alpha = m + s(n-1) + t$, where $0 \leqslant s \leqslant m$ and $0 \leqslant t < n - 1$. Let $\Sigma = \{a, b, c\}$.

Define an m-state NFA $A = (Q_A, \Sigma, \delta_A, q_0, F_A)$, where $Q_A = \{q_0, q_1, \dots, q_{m-1}\}$, $F_A = \{q_0\}$ and for any $i \in \{0, 1, \dots, m-1\}$ and any $X \in \Sigma$,

$$\delta_A(q_i, X) = \begin{cases} \{q_{(i+1) \bmod m}\}, & \text{if } X = a, \\ \{q_i\}, & \text{if } i \leqslant s-1 \text{ and } X = b, \\ \{q_i\}, & \text{if } i = s \text{ and } X = c, \\ \emptyset, & \text{otherwise.} \end{cases}$$

Define an n-state DFA $B = (Q_B, \Sigma, \delta_B, p_0, F_B)$, where $Q_B = \{p_0, p_1, \dots, p_{n-1}\}$, $F_B = \{p_0\}$, and for any $i \in \{0, 1, \dots, n-1\}$ and any $X \in \Sigma$,

$$\delta_B(p_i, X) = \begin{cases} \{p_0\}, & \text{if } i = 0 \text{ and } X = a, \\ \{p_{(i+1) \bmod n}\}, & \text{if } X = b, \\ \{p_{i+1}\}, & \text{if } i \leqslant t-1 \text{ and } X = c, \\ \{p_0\}, & \text{if } i = t \text{ and } X = c, \\ \emptyset, & \text{otherwise.} \end{cases}$$

The NFA A and B are shown in Fig. 4. Both automata are minimal since the sets of pairs of strings $\{(a^i, a^{m-i}) \mid i = 0, 1, \dots, m-1\}$ and $\{(b^i, b^{n-i}) \mid i = 0, 1, \dots, n-1\}$ are fooling sets for the languages $L(A)$ and $L(B)$, respectively.

Let $C = (Q_A \times Q_B, \Sigma, \delta, [q_0, p_0], F)$, be the cross-product of automata A and B accepting the language $L(A) \cap L(B)$. Let $\mathcal{R}$ be the following set of states.

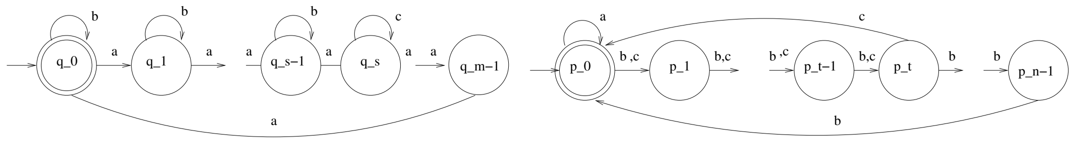

Fig. 4. The nondeterministic finite automata A and B.

$\mathcal{R} = \{[q_i, p_0] \mid 0 \leqslant i < m\} \cup \{[q_i, p_j] \mid 0 \leqslant i < s, 1 \leqslant j < n\} \cup \{[q_s, p_j] \mid 1 \leqslant j \leqslant t\}$. Any state in $\mathcal{R}$ is reachable in the NFA C because we have $[q_i, p_0] \in \delta([q_0, p_0], a^i)$ for $i = 0, 1, \ldots, m-1$, $[q_i, p_j] \in \delta([q_0, p_0], a^i b^j)$ for $i = 0, 1, \ldots, s-1$ and $j = 1, 2, \ldots, n-1$, and $[q_s, p_j] \in \delta([q_0, p_0], a^s c^j)$ for $j = 1, 2, \ldots, t$. Next, no other state in $Q_A \times Q_B$ is reachable in the NFA C because the initial state $[q_0, p_0]$ of the NFA C is in $\mathcal{R}$ and for any state $[q, p]$ in $\mathcal{R}$ and any symbol X in Σ either $\delta([q, p], X) \subseteq \mathcal{R}$ or $\delta([q, p], X) = \emptyset$. Thus the language $L(A) \cap L(B)$ is accepted by an $(m + s(n-1) + t)$-state NFA. To prove the lemma consider the following sets of pairs of strings:

$\mathcal{A} = \{(a^i, a^{m-i}) \mid i = 0, 1, \ldots, m-1\}$,

$\mathcal{B}_k = \{(a^k b^i, b^{n-i} a^{m-k}) \mid i = 1, 2, \ldots, n-1\}$, for $k = 0, 1, \ldots, s-1$,

$\mathcal{C} = \{(a^s c^i, c^{t+1-i} a^{m-s}) \mid i = 1, 2, \ldots, t\}$.

Let $\mathcal{D} = \mathcal{A} \cup \mathcal{B}_0 \cup \mathcal{B}_1 \cup \cdots \cup \mathcal{B}_{s-1} \cup \mathcal{C}$. We will show that the set $\mathcal{D}$ is a fooling set for the language $L(A) \cap L(B)$. We need to show that

(1) for any pair (x_i, y_i) in $\mathcal{D}$, the string $x_i y_i$ is in the language $L(A) \cap L(B)$, and

(2) for any two different pairs (x_i, y_i) and (x_j, y_j) in $\mathcal{D}$, at least one of the strings $x_i y_j$ and $x_j y_i$ is not in the language $L(A) \cap L(B)$.

To prove (1) note that the strings a^m, $a^k b^n a^{m-k}$ $(k = 0, 1, \ldots, s-1)$, and $a^s c^{t+1} a^{m-s}$ are in the language $L(A) \cap L(B)$.

To prove (2) we have six cases to consider:

(i) Both pairs are in $\mathcal{A}$. Let $0 \leqslant i < j \leqslant m-1$. Then the string $a^i a^{m-j}$ is not in the language $L(A)$ since $0 < m - j + i < m$.

(ii) Both pairs are in $\mathcal{B}_k$ for some k $(0 \leqslant k \leqslant s-1)$. Let $1 \leqslant i < j \leqslant n-1$. Then the string $a^k b^i b^{n-j} a^{m-k}$ is not in the language $L(B)$ since $0 < n - j + i < n$.

(iii) Both pairs are in $\mathcal{C}$. Let $1 \leqslant i < j \leqslant t$. Then the string $a^s c^i c^{t+1-j} a^{m-s}$ is not in the language $L(B)$ since $0 < t + 1 - j + i < t + 1$.

(iv) The first pair is in $\mathcal{A}$ and the second in some $\mathcal{B}_k$ (or in $\mathcal{C}$). Then the string $a^k b^i a^{m-j}$ (the string $a^s c^i a^{m-j}$, respectively) is not in the language $L(B)$ since $1 \leqslant i \leqslant n-1$ $(1 \leqslant i \leqslant t$, respectively).

(v) The first pair is in some $\mathcal{B}_k$ and the second in some $\mathcal{B}_l$ with $0 \leqslant k < l \leqslant s-1$. Then the string $a^k b^i b^{n-j} a^{m-l}$ is not in the language $L(A)$ since $0 < k + m - l < m$.

(vi) The first pair is in some $\mathcal{B}_k$ and the second in $\mathcal{C}$. Then the string $a^s c^i b^{n-j} a^{m-k}$ is not in the language $L(A)$ since $i \geqslant 1$ and $n - j \geqslant 1$.

Thus we have shown that the set $\mathcal{D}$ is a fooling set for the language $L(A) \cap L(B)$. By Lemma 5, any NFA for the language $L(A) \cap L(B)$ needs at least $m + s(n-1) + t$ states and our proof is complete. ☐

As a corollary of the two lemmata above, we get the following result.

Theorem 4. *For any positive integers m, n, α such that $1 \leqslant \alpha \leqslant mn$, there exist a minimal NFA A of m states and a minimal NFA B of n states such that minimal NFAs for the language $L(A) \cap L(B)$ have exactly α states.* □

6 Conclusions

In this paper, we investigated the state complexity and the nondeterministic state complexity of union and intersection of regular languages. In the deterministic case, we showed that the entire range of complexities between 1 and mn can be obtained by the union (intersection, respectively) of an m-state DFA language and an n-state DFA language for any integers m and n such that $m \geqslant 2$ and $n \geqslant 2$. Next, we proved that the nondeterministic state complexity of the union of an m-state NFA language and an n-state NFA language may be arbitrary between 1 and $m + n + 1$, except for the case of $m = 1$ and $n = 1$. To prove these results we used a binary alphabet. In the case of a unary alphabet, similar results probably do not hold. Finally, we showed that the nondeterministic state complexity of the intersection of an m-state NFA language and an n-state NFA language may be arbitrary between 1 and mn. We proved the last result for a ternary alphabet. The question whether this result holds likewise for a binary alphabet remains open.

References

1. J. C. Birget, Intersection and union of regular languages and state complexity, *Inform. Process. Lett.* **43** (1992) 185–190.
2. J. C. Birget, Partial orders on words, minimal elements of regular languages, and state complexity, *Theoret. Comput. Sci.* **119** (1993) 267–291.
3. C. Câmpeanu, K. Culik II, K. Salomaa, and S. Yu, State complexity of basic operations on finite languages, in: O. Boldt, H. Jürgensen (Eds.), *Proc. 4th International Workshop on Implementing Automata (WIA'99),* LNCS 2214, Springer-Verlag, Heidelberg, 2001, pp. 60–70.
4. C. Câmpeanu, K. Salomaa, and S. Yu, Tight lower bound for the state complexity of shuffle of regular languages, *J. Autom. Lang. Comb.* **7** (2002) 303–310.
5. M. Domaratzki, State complexity and proportional removals, *J. Autom. Lang. Comb.* **7** (2002) 455–468.
6. K. Ellul, B. Krawetz, J. Shallit, and M. W. Wang, Regular expressions: new results and open problems, *J. Autom. Lang. Comb.*, to appear.
7. I. Glaister and J. Shallit, A lower bound technique for the size of nondeterministic finite automata, *Inform. Process. Lett.* **59** (1996) 75–77.
8. M. Holzer and M. Kutrib, Nondeterministic descriptional complexity of regular languages, *Internat. J. Found. Comput. Sci.* **14** (2003) 1087-1102.
9. J. Hromkovič, *Communication Complexity and Parallel Computing,* Springer-Verlag, Berlin, Heidelberg, 1997.
10. J. Hromkovič, Descriptional complexity of finite automata: concepts and open problems, *J. Autom. Lang. Comb.* **7** (2002) 519–531.
11. G. Jirásková, State complexity of some operations on binary regular languages, *Theoret. Comput. Sci.* **330** (2005) 287–298.
12. E. Leiss, Succinct representation of regular languages by boolean automata, *Theoret. Comput. Sci.* **13** (1981) 323–330.
13. O. B. Lupanov, A comparison of two types of finite automata, *Problemy Kibernetiki,* **9** (1963) 321-326 (in Russian).
14. Yu. I. Lyubich, Estimates for optimal determinization of nondeterministic autonomous automata, *Sibirskii Matematichskii Zhurnal,* **5** (1964) 337-355 (in Russian).

15. A. N. Maslov, Estimates of the number of states of finite automata, *Dokl. Akad. Nauk SSSR,* **194** (1970) 1266–1268 (in Russian). English translation: *Soviet Math. Dokl.* **11** (1970) 1373–1375.
16. F. R. Moore, On the bounds for state-set size in the proofs of equivalence between deterministic, nondeterministic, and two-way finite automata, *IEEE Trans. Comput.* **20** (1971) 1211–1214.
17. G. Pighizzini and J. Shallit, Unary language operations, state complexity and Jacobsthal's function, *Internat. J. Found. Comput. Sci.* **13** (2002) 145–159.
18. M. Rabin and D. Scott, Finite automata and their decision problems, *IBM Res. Develop.* **3** (1959) 114–129.
19. A. Salomaa, D. Wood, and S. Yu, On the state complexity of reversals of regular languages, *Theoret. Comput. Sci.* **320** (2004) 315–329.
20. M. Sipser, *Introduction to the theory of computation,* PWS Publishing Company, Boston, 1997.
21. S. Yu, Q. Zhuang, and K. Salomaa, The state complexity of some basic operations on regular languages, *Theoret. Comput. Sci.* **125** (1994) 315–328.
22. S. Yu, Chapter 2: Regular languages, in: G. Rozenberg, A. Salomaa, (Eds.), *Handbook of Formal Languages - Vol. I,* Springer-Verlag, Berlin, New York, 1997, pp. 41–110.
23. S. Yu, A renaissance of automata theory? *Bull. Eur. Assoc. Theor. Comput. Sci. EATCS* **72** (2000) 270–272.
24. S. Yu, State complexity of regular languages, *J. Autom. Lang. Comb.* **6** (2001) 221–234.
25. S. Yu, State complexity of finite and infinite regular languages, *Bull. Eur. Assoc. Theor. Comput. Sci. EATCS* **76** (2002) 142–152.