COMBINED OPERATIONS ON PREFIX-FREE AND SUFFIX-FREE LANGUAGES

Hae-Sung Eom$^{(A)}$ Matúš Palmovský$^{(B)}$

$^{(A)}$Department of Computer Science, Yonsei University
50, Yonsei-Ro, Seodaemun-Gu, Seoul 120-749, Korea
haesung@cs.yonsei.ac.kr

$^{(B)}$Mathematical Institute, Slovak Academy of Science, Grešákova 6, 040 01 Košice, Slovakia
palmovsky@saske.sk

Abstract
We investigate the state complexity of combined operations for prefix-free and suffix-free regular languages. Prefix-free and suffix-free deterministic finite-state automata have some special properties that are crucial for obtaining the precise state complexity of basic operations. Based on these properties, we establish the state complexity of several operations: catenation-of-union, catenation-of-intersection, catenation-of-star.

1. Introduction

Given a regular language L, the state complexity of L is the number of states in the minimal deterministic finite-state automaton (DFA) for L. The state complexity of an operation on regular languages is the number of states that are necessary and sufficient in the worst-case for a DFA to accepts the language obtained from the operation. Maslov ([15]) obtained the state complexity of catenation and later Yu et al. ([21]) investigated the operational complexity further. The state complexity of an operation is calculated based on the structural properties of the input regular languages and a given operation. Many applications using regular languages require finite-state automata (FAs) of very large size. This makes the estimated upper bound of the state complexity useful in practice since it helps to manage resources efficiently. Moreover, it is a challenging quest to verify whether or not an estimated upper bound can be reached.

Yu ([20]) gave a comprehensive survey of the state complexity of regular languages. Salomaa et al. ([18]) studied classes of languages for which the reversal operation reaches an exponential upper bound. As special cases of the state complexity, researchers examined the state complexity of finite languages ([2, 7]), the state complexity of unary language operations ([17]) and

$^{(B)}$Research supported by VEGA grant 2/0084/15 and grant APVV-15-0091. The work on this paper was done during a stay of the second author at the university *Department of Computer Science, Yonsei University.*

the nondeterministic descriptional complexity of regular languages ([10]). For regular language codes, Han et al. ([9]) studied the state complexity of prefix-free regular languages. Similarly, based on suffix-freeness, Han and Salomaa ([8]) looked at the state complexity of suffix-free regular languages. There are several other results with respect to the state complexity of different operations ([3, 5, 6, 11, 12, 16]).

In this paper we study the state complexity of several operations combined with catenation in the classes of prefix-free and suffix-free languages. We use unique structural properties of prefix-free and suffix-free deterministic automata to get tight upper bounds for union, intersection, and star combined with catenation. The paper is organized as follows. In Section 2, we define some basic notions and state some preliminary results. Then we present the state complexities of four combined operations in the following sections. We compare the state complexity of basic operations and the state complexity of combined operations for prefix-free and suffix-free regular languages, and conclude the paper in Section 7.

2. Preliminaries

We assume that the reader is familiar with basic notions in formal languages and automata theory, and for complete background knowledge, we refer to Wood [19].

Let Σ denote a finite alphabet of characters and Σ^* denote the set of all strings over Σ. The size $|\Sigma|$ of Σ is the number of characters in Σ. A language over Σ is any subset of Σ^*. The symbol $\emptyset$ denotes the empty language and the symbol λ denotes the null string. For a finite set A, we denote by $|A|$ its size and by 2^A its power set. For strings x, y, and z such that $z = xy$, we say that x is a *prefix* of z and y is a *suffix* of z. We define a language L to be prefix (suffix)-free if for any two distinct strings x and y in L, x is not a prefix (suffix) of y.

A *deterministic finite automaton* (DFA) A is specified by a tuple $(Q, \Sigma, \delta, s, F)$, where Q is a finite set of states, Σ is an input alphabet, $\delta : Q \times \Sigma \to Q$ is a transition function, $s \in Q$ is the start state and $F \subseteq Q$ is a set of final states. The transition function δ can be extended to the domain $Q \times \Sigma^*$ in the natural way. Given a DFA A, we assume that A is complete; namely, for each state q and each letter a, the transition $\delta(q, a)$ is defined. However, in some constructions, we also use incomplete DFAs, in which δ is a partial function. A complete DFA may have a sink state, that is a state from which no string is accepted. We assume that a DFA has at most one sink state since all sink states are equivalent. For a transition $\delta(p, a) = q$ in A, we say that p has an *out-transition* and q has an *in-transition*. Furthermore, p is a *source state* of q and q is a *target state* of p. We say that A is *non-returning* if the start state of A does not have any in-transitions. A string x over Σ is accepted by A if $\delta(s, w) \in F$. The language $L(A)$ of A is the set of all strings that are accepted by A. We define a state q of A to be *reachable* if there is a path from the start state to q. Two states p and q are distinguishable if there is a string w such that exactly one of the states $\delta(p, w)$ and $\delta(q, w)$ is final.

The *state complexity* $\mathcal{SC}(L)$ of a regular language L is defined to be the size of the minimal (with respect to the number of states) DFA recognizing L.

It is well-known that a minimal DFA for a prefix-free language has a sink state and exactly one final state, from which all the transitions go to the sink state. Next, a minimal DFA for a suffix-free language must be non-returning.

We recall a known result that is useful to tackle the state complexity problem for suffix-free regular languages.

Lemma 2.1 (Cmorik and Jirásková [4]) *Let A be a non-returning DFA with a sink state and a unique final state. If no two distinct states of A go to a non-sink state by the same symbol, then $L(A)$ is suffix-free.*

A *nondeterministic finite automaton* (NFA) is a tuple $A = (Q, \Sigma, \delta, I, F)$, where Q is a finite state set, Σ is a finite input alphabet, $\delta \colon Q \times \Sigma \to 2^Q$ is the transition function that can be extended to the domain $2^Q \times \Sigma^*$, $I \subseteq Q$ is the set of initial states, and $F \subseteq Q$ is the set of final states. If $q \in \delta(p, a)$, then we say that (p, a, q) is a transition in A. The language accepted by the NFA A is the set of strings $L(A) = \{w \in \Sigma^* \mid \delta(I, w) \cap F \neq \emptyset\}$. Every NFA $A = (Q, \Sigma, \delta, I, F)$ can be converted to an equivalent DFA $A' = (2^Q, \Sigma, \delta', I, F')$, where $F' = \{S \in 2^Q \mid S \cap F \neq \emptyset\}$ and $\delta'(S, a) = \delta(S, a)$ for each S in 2^Q and each a in Σ. We call the DFA A' the *subset automaton* of NFA A. The subset automaton may not be minimal since some of its states may be unreachable or equivalent to some other states.

A state q of NFA A is *uniquely distinguishable* [1] if there is a string w in Σ^* which is accepted by A from and only from state q, that is, we have $\delta(p, w) \in F$ if and only if $p = q$. We also say that q is *uniquely distinguishable by the string* w. Next, we say that (p, a, q) is a *unique in-transition* on a going to q, if there is no state r in Q such that $r \neq p$ and $q \in \delta(r, a)$. Finally, we say that a state q is *uniquely reachable* from p if $p = p_0 \xrightarrow{a_1} p_1 \xrightarrow{a_2} p_2 \xrightarrow{a_3} \cdots \xrightarrow{a_k} p_k = q$, and each transition (p_{i-1}, a_i, p_i) is a unique in-transition on a_i going to p_i.

In [1], the following sufficient conditions for an NFA N, under which the subset automaton of N does not have equivalent states, are stated. For the sake of completeness, we recall their proofs here.

Proposition 2.2 *If each state of an NFA A is uniquely distinguishable, then the subset automaton of A does not have equivalent states.*

Proof. Let S and T be two distinct subsets of the subset automaton. Then there is a state q in Q such that $q \in S \setminus T$. Since q is uniquely distinguishable, there is a string w with $\delta(p, w) \in F$ if and only if $p = q$. Then w is accepted from S and rejected from T. □

Proposition 2.3 *Let q be uniquely distinguishable and (p, a, q) be a unique in-transition on a going to state q. Then p is uniquely distinguishable.*

Proof. Let q be uniquely distinguishable by w. Then the string aw is accepted from and only from p, so p is uniquely distinguishable. □

Proposition 2.4 *Let $G(N)$ be a subgraph of unique in-transitions of an NFA N. Let a uniquely distinguishable state of N be reachable from each state of N in the subgraph $G(N)$. Then the subset automaton of N does not have equivalent states.*

Proof. If a uniquely distinguishable state is reached from a state p in $G(N)$, then p is uniquely distinguishable by Proposition 2.3. Hence each state of N is uniquely distinguishable. By Proposition 2.2, the subset automaton of N does not have equivalent states. □

3. State Complexity of $L_1 \cdot (L_2 \cup L_3)$

We start with the state complexity of $L_1 \cdot (L_2 \cup L_3)$. Our aim is to show that for prefix-free languages, the tight upper bound is $m + np - 4$ and for suffix-free languages, the tight upper bound is $(m-1)2^{n+p-4} + 1$, where $\mathcal{SC}(L_1) = m$, $\mathcal{SC}(L_2) = n$, and $\mathcal{SC}(L_3) = p$. Our worst-case examples are defined over a three-letter alphabet in the prefix-free case and over a six-letter alphabet in the suffix-free case.

Theorem 3.1 *Let $m, n, p \geq 3$ and L_1, L_2 and L_3 be regular prefix-free languages over an alphabet Σ with $\mathcal{SC}(L_1) = m$, $\mathcal{SC}(L_2) = n$, and $\mathcal{SC}(L_3) = p$. Then $\mathcal{SC}(L_1 \cdot (L_2 \cup L_3)) \leq m + np - 4$, and the bound is tight if $|\Sigma| \geq 3$.*

Proof. Let $A_1 = (Q_1, \Sigma, \delta_1, s_1, \{f_1\})$, $A_2 = (Q_2, \Sigma, \delta_2, s_2, \{f_2\})$, and $A_3 = (Q_3, \Sigma, \delta_3, s_3, \{f_3\})$ be minimal DFAs for L_1, L_2, and L_3, respectively, with sink states d_1, d_2, and d_3. Construct an NFA N for $L_1 \cdot (L_2 \cup L_3)$ from DFAs A_1, A_2, and A_3 as follows:
(1) omit states f_1, d_1, d_2, d_3 and all the transitions going to or from these states;
(2) merge states f_2 and f_3 into a new state f;
(3) for each transition (q, a, f_1) in A_1 add two transitions (q, a, s_2) and (q, a, s_3);
(4) the initial state of N is s_1, and the set of final states is $\{f\}$;
see Figure 1 for an example.

Since A_1, A_2, A_3 are deterministic, in the subset automaton of N, only the following sets may be reachable:

- $\{q\}$, where $q \in Q_1 \setminus \{f_1, d_1\}$;
- $\{r, t\}, \{r, f\}, \{r\}, \{t, f\}, \{t\}$, where $r \in Q_2 \setminus \{f_2, d_2\}$ and $t \in Q_3 \setminus \{f_3, d_3\}$;
- $\{f\}$, and the empty set.

In total we get at most $(m-2) + (n-2)(p-2) + 2(n-2) + 2(p-2) + 2 = m + np - 4$ reachable subsets. This proves the upper bound.

For tightness, consider the languages L_1, L_2, and L_3 accepted by DFAs A_1, A_2, and A_3 shown in Figure 1 (top); to keep our figures transparent, we do not display the sink states anywhere. Construct an NFA N as described above; see Figure 1 (bottom). Then in the subset automaton of N, the initial subset is $\{q_0\}$, and for each i, j, k with $0 \leq i \leq m-3$, $0 \leq j \leq n-3$, $0 \leq k \leq p-3$, we have

- $\{q_0\} \xrightarrow{a^i} \{q_i\}$;
- $\{q_{m-3}\} \xrightarrow{a} \{r_0, t_0\} \xrightarrow{b^j c^k} \{r_j, t_k\}$;
- $\{r_j, t_{p-3}\} \xrightarrow{c} \{r_j, f\} \xrightarrow{c} \{r_j\}$; $\{r_{n-3}, t_k\} \xrightarrow{b} \{f, t_k\} \xrightarrow{b} \{t_k\}$; and $\{r_{n-3}\}xrightarrowb\{f\} \xrightarrow{b}$ $\emptyset$.

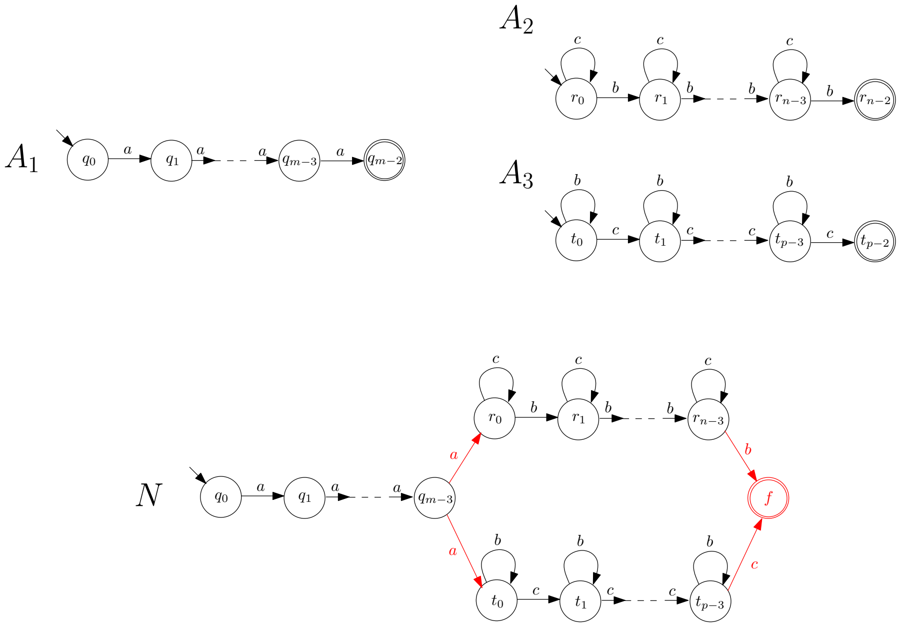
A_2
r_0 r_1 r_{n-3} r_{n-2}
A_1
q_0 q_1 q_{m-3} q_{m-2}
A_3
t_0 t_1 t_{p-3} t_{p-2}
N
f
a b c

Figure 1: Prefix-free witnesses (top) and NFA N (bottom) for $L_1 \cdot (L_2 \cup L_3)$.

Thus all $m+np-4$ subsets are reachable. To prove distinguishability, notice that each transition in N is a unique in-transition, and that the unique final state f is reachable from each state in N. By Proposition 2.4, the subset automaton of N does not have equivalent states. □

Theorem 3.2 *Let $m, n, p \geq 3$, and L_1, L_2, L_3 be regular suffix-free languages over an alphabet Σ with $\mathcal{SC}(L_1) = m$, $\mathcal{SC}(L_2) = n$, and $\mathcal{SC}(L_3) = p$. Then $\mathcal{SC}(L_1 \cdot (L_2 \cup L_3)) \leq (m-1)2^{n+p-4}+1$, and the bound is tight if $|\Sigma| \geq 6$.*

Proof. Let L_1, L_2, L_3 be suffix-free languages accepted by minimal DFAs $A_1 = (Q_1, \Sigma, \delta_1, s_1, F_1)$, $A_2 = (Q_2, \Sigma, \delta_2, s_2, F_2)$, and $A_3 = (Q_3, \Sigma, \delta_3, s_3, F_3)$, with sink states d_1, d_2, d_3, respectively. Then A_1, A_2, A_3 are non-returning. Construct an NFA N for $L_1 \cdot (L_2 \cup L_3)$ from DFAs A_1, A_2, and A_3 as follows:

(1) omit states d_1, s_2, d_2, s_3, d_3 and all transitions going to these states;
(2) for each symbol a in Σ and each state q in F_1,
 (a) if $(\delta_2(s_2, a) \neq d_2)$, then add the transition $(q, a, \delta_2(s_2, a))$;
 (b) if $(\delta_3(s_3, a) \neq d_3)$, then add the transition $(q, a, \delta_3(s_3, a))$;
(3) the initial state of N is s_1, and the set of final states is $F_2 \cup F_3$.

In the corresponding subset automaton, only the following sets may be reachable:

- $\{s_1\}$;
- $\{q\} \cup S$ and S, where $q \in Q_1 \setminus \{s_1, d_1\}$ and $S \subseteq (Q_2 \setminus \{s_2, d_2\}) \cup (Q_3 \setminus \{s_3, d_3\})$.

This gives at most $(m-1)2^{n+p-4}+1$ reachable states, and proves the upper bound.

For tightness, consider the languages L_1, L_2, and L_3 accepted by DFAs A_1, A_2, and A_3 shown in Figure 2. By Lemma 2.1, L_1, L_2, L_3 are suffix-free. Construct the NFA N for $L_1 \cdot (L_2 \cup L_3)$ as described above; see Figure 3. Then in the subset automaton of N, the initial subset is $\{q_0\}$, and we have $\{q_0\} \xrightarrow{af^{m-3}} \{q_{m-2}\} \xrightarrow{a^{n+p}} \{q_{m-2}\} \cup \{r_1, \ldots, r_{n-2}\} \cup \{t_1, \ldots, t_{p-2}\}$.

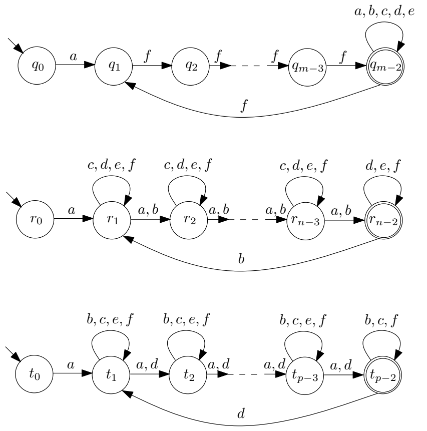

Figure 2: Suffix-free witnesses for $L_1 \cdot (L_2 \cup L_3)$.

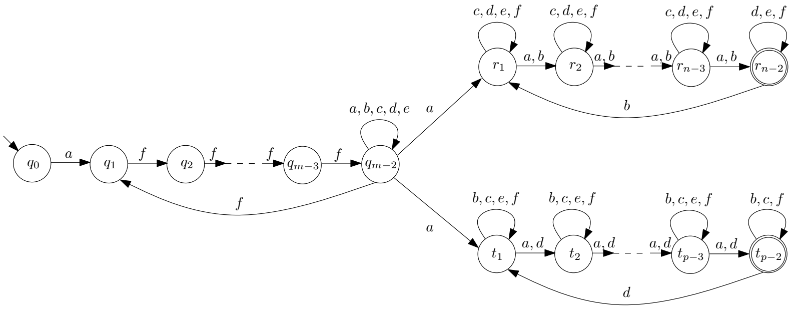

Figure 3: The NFA N for $L_1 \cdot (L_2 \cup L_3)$, where L_1, L_2, L_3 are accepted by the DFAs from Figure 2.

Now notice that using b we can shift any subset of $\{r_1, \ldots, r_{n-2}\}$ cyclically by one, while using c we can eliminate state r_{n-2} from any subset containing r_{n-2}. It follows that each subset R of $\{r_1, \ldots, r_{n-2}\}$ can be reached from $\{r_1, \ldots, r_{n-2}\}$ by a string u_R over $\{b, c\}$. Moreover, we have a loop on b, c in q_{m-2} and in each state t_k. Thus we get

$$\{q_{m-2}\} \cup \{r_1, \ldots, r_{n-2}\} \cup \{t_1, \ldots, t_{p-2}\} \xrightarrow{u_R} \{q_{m-2}\} \cup R \cup \{t_1, \ldots, t_{p-2}\}.$$

Symmetrically, we can remove states from $\{t_1, \ldots, t_{p-2}\}$ using d, e, and reach every set T by a string v_T over $\{d, e\}$, so $\{q_{m-2}\} \cup R \cup \{t_1, \ldots, t_{p-2}\} \xrightarrow{v_T} \{q_{m-2}\} \cup R \cup T$. Next, we have $\{q_{m-2}\} \cup R \cup T \xrightarrow{f^i} \{q_i\} \cup R \cup T$, and

$$\{q_{m-2}\} \cup \{r_1, \ldots, r_{n-2}, t_1, \ldots, t_{p-2}\} \xrightarrow{f} \{q_1\} \cup \{r_1, \ldots, r_{n-2}, t_1, \ldots, t_{p-2}\} \xrightarrow{b}$$
$$\{r_1, \ldots, r_{n-2}\} \cup \{t_1, \ldots, t_{p-2}\} \xrightarrow{u_R v_T} R \cup T.$$

This gives $(m-1)2^{n+p-4} + 1$ reachable states. To get distinguishability, notice that the states r_{n-2} and t_{p-2} are uniquely distinguishable in N since e is accepted from and only from r_{n-2} and c is accepted from and only from t_{p-2}. Next, in the subgraph $G(N)$ given by unique in-transitions (q_0, a, q_1), (q_i, f, q_{i+1}) with $1 \le i \le m-3$, (q_{m-2}, a, r_1), (r_j, a, r_{j+1}) with $1 \le j \le n-3$, (q_{m-2}, a, t_1), (t_k, a, t_{k+1}) with $1 \le k \le p-3$, either r_{n-2} or t_{p-2} can be reached from every state of N. By Proposition 2.4, the subset automaton of N does not have equivalent states. □

4. State Complexity of $(L_1 \cup L_2) \cdot L_3$

Now we consider the state complexity of $(L_1 \cup L_2) \cdot L_3$. We get tight upper bounds for prefix-free and suffix-free regular languages L_1, L_2, and L_3. To prove tightness, we use a binary alphabet in the prefix-free case, and a six-letter alphabet in the suffix-free case.

Theorem 4.1 *Let $m, n, p \geq 3$, and L_1, L_2, L_3 be prefix-free languages over Σ with $\mathcal{SC}(L_1) = m$, $\mathcal{SC}(L_2) = n$, and $\mathcal{SC}(L_3) = p$. Then $\mathcal{SC}((L_1 \cup L_2) \cdot L_3) \leq (m-2)\ (n-2) + (m+n-4)p + (p^2 - p + 2)/2$, and the bound is tight if $|\Sigma| \geq 2$.*

Proof. Let $A_1 = (Q_1, \Sigma, \delta_1, s_1, \{f_1\})$, $A_2 = (Q_2, \Sigma, \delta_2, s_2, \{f_2\})$, and $A_3 = (Q_3, \Sigma, \delta_3, s_3, \{f_3\})$ be minimal DFAs for prefix-free languages L_1, L_2, and L_3, respectively, with sink states d_1, d_2, and d_3. Construct an NFA N for $(L_1 \cup L_2) \cdot L_3$ from DFAs A_1, A_2, and A_3 as follows:
(1) omit states f_1, d_1, f_2, d_2, d_3 and all transitions going to or from these states;
(2) if $(q, a, f_1) \in \delta_1$, then add (q, a, s_3);
(3) if $(r, a, f_2) \in \delta_2$, then add (r, a, s_3);
(4) the set of initial states is $\{s_1, s_2\}$ and the set of final states is $\{f_3\}$;
see Figure 4 for an example. In the subset automaton of N, only the following sets may be reachable:

- $\{q, r\}$, where $q \in Q_1 - \{f_1, d_1\}$ and $r \in Q_2 - \{f_2, d_2\}$;
- $\{q, t\}$ and $\{q\}$, where $q \in Q_1 - \{f_1, d_1\}$ and $t \in Q_3 - \{d_3\}$;
- $\{r, t\}$ and $\{r\}$, where $r \in Q_2 - \{f_2, d_2\}$ and $t \in Q_3 - \{d_3\}$;
- $\{t, t'\}$ and $\{t\}$ and the empty set for $t, t' \in Q_3 - \{d_3\}$.

In total we get at most $(m-2)(n-2) + (m-2)p + (n-2)p + (p-1)(p-2)/2 + (p-1) + 1 = (m-2)(n-2) + (m+n-4)p + (p^2 - p + 2)/2$ reachable subsets, which proves the upper bound.

To prove tightness, consider binary prefix-free languages L_1, L_2, L_3 accepted by DFAs A_1, A_2, A_3 shown in Figure 4 (top).

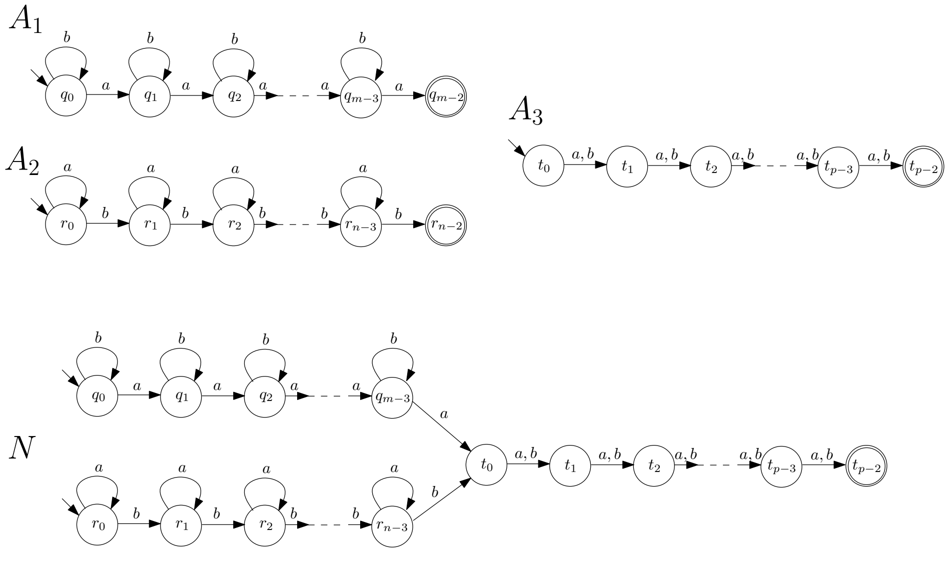

Figure 4: Prefix-free witnesses (top) and NFA N (bottom) for $(L_1 \cup L_2) \cdot L_3$.

Construct the NFA N for $(L_1 \cup L_2) \cdot L_3$ as described above; see Figure 4 (bottom). Then in the subset automaton of N, the initial states are q_0 and r_0, and for each i, j, k, ℓ with $0 \leq i \leq m-3$, $0 \leq j \leq n-3$, and $0 \leq k < \ell \leq p-2$,

- $\{q_0, r_0\} \xrightarrow{a^i b^j} \{q_i, r_j\}$;
- $\{q_i, r_{n-3}\} \xrightarrow{b} \{q_i, t_0\} \xrightarrow{b^k} \{q_i, t_k\}$, and $\{q_i, t_{p-2}\} \xrightarrow{b} \{q_i\}$;
- $\{r_j, q_{m-3}\} \xrightarrow{a} \{r_j, t_0\} \xrightarrow{a^k} \{r_j, t_k\}$, and $\{r_j, t_{p-2}\} \xrightarrow{a} \{r_j\}$;
- $\{q_{m-3}, t_0\} \xrightarrow{b^{\ell-k-1}} \{q_{m-3}, t_{\ell-k-1}\} \xrightarrow{a} \{t_0, t_{\ell-k}\} \xrightarrow{a^k} \{t_k, t_\ell\}$;
- $\{q_{m-3}\} \xrightarrow{a} \{t_0\} \xrightarrow{a^k} \{t_k\}$ and $\{t_{p-2}\} \xrightarrow{a} \emptyset$.

This proves the reachability of $(m-2)(n-2) + (m+n-4)p + (p^2 - p + 2)/2$ subsets. To prove distinguishability, notice that each transition in N is a unique in-transition, and that the unique final state state t_{p-2} is reachable from each state in N. By Proposition 2.4, the subset automaton of N does not have equivalent states. □

Theorem 4.2 *Let $m, n, p \geq 4$, and L_1, L_2, L_3 be suffix-free languages over an alphabet Σ with $\mathcal{SC}(L_1) = m$, $\mathcal{SC}(L_2) = n$, and $\mathcal{SC}(L_3) = p$. Then $\mathcal{SC}((L_1 \cup L_2) \cdot L_3) \leq (m-1)(n-1)2^{p-2} + 1$, and the bound is tight if $|\Sigma| \geq 6$.*

Proof. Let L_1, L_2, L_3 be suffix-free languages accepted by minimal DFAs $A_1 = (Q_1, \Sigma, \delta_1, s_1, F_1)$, $A_2 = (Q_2, \Sigma, \delta_2, s_2, F_2)$, and $A_3 = (Q_3, \Sigma, \delta_3, s_3, F_3)$, with sink states d_1, d_2, d_3, respectively. Then A_1, A_2, A_3 are non-returning. Construct an NFA N for $(L_1 \cup L_2) \cdot L_3$ from DFAs A_1, A_2, and A_3 as follows:

(1) omit states d_1, d_2, s_3, d_3 and all the transitions going to or from these states;
(2) for each a in Σ and each q in $F_1 \cup F_2$, if $\delta_3(s_3, a) \neq d_3$, then add the transition $(q, a, \delta_3(s_3, a))$;
(3) the set of initial states of N is $\{s_1, s_2\}$ and the set of final states is F_3;

see Figure 5 for an example. Since A_1, A_2, A_3 are deterministic and non-returning, in the subset automaton of N, only the following sets may be reachable:

- $\{s_1, s_2\}$;
- $\{q, r\} \cup T$, where $q \in Q_1 \setminus \{s_1, d_1\}$, $r \in Q_2 \setminus \{s_2, d_2\}$, and $T \subseteq Q_3 \setminus \{s_3, d_3\}$;
- $\{q\} \cup T$, where $q \in Q_1 \setminus \{s_1, d_1\}$ and $T \subseteq Q_3 \setminus \{s_3, d_3\}$;
- $\{r\} \cup T$, where $r \in Q_2 \setminus \{s_2, d_2\}$ and $T \subseteq Q_3 \setminus \{s_3, d_3\}$;
- T, where $T \subseteq Q_3 \setminus \{s_3, d_3\}$.

In total, we get at most $1 + (m-2)(n-2)2^{p-2} + (m-2)2^{p-2} + (n-2)2^{p-2} + 2^{p-2} = (m-1)(n-1)2^{p-2} + 1$ reachable states. This proves the upper bound.

To prove tightness, let L_1, L_2, L_3 be the languages accepted by DFAs A_1, A_2, A_3 shown in Figure 5 (top). By Lemma 2.1, the languages L_1, L_2, L_3 are suffix-free since we have $m, n \geq 4$. Construct the NFA N as described above, that is, remove the states q_{m-1}, r_{n-1}, 0, $p-1$, and add the transitions $(q_{m-2}, a, 1)$ and $(r_{n-2}, a, 1)$; see Figure 5 (bottom). First, let us show that for each $T \subseteq \{1, 2, \ldots, p-2\}$, the set $\{q_{m-2}, p_{n-2}\} \cup T$ is reachable in the subset automaton of N. The proof is by induction on $|T|$. The basis, with $|T| = 0$, holds true since $\{q_{m-2}, p_{n-2}\}$ is reached from the initial state $\{q_0, r_0\}$ by $ac^{m-3}d^{n-3}$. Next, each set $\{q_{m-2}, r_{n-2}, k_1, k_2, k_3, \ldots, k_\ell\}$, where $1 \leq \ell \leq p-2$ and $1 \leq k_1 < k_2 < k_3 < \cdots < k_\ell <= p-2$, is reached from the set $\{q_{m-2}, r_{n-2}, k_2 - k_1, k_3 - k_1, \ldots, k_\ell - k_1\}$ by $ab^{k_1 - 1}$. This proves our claim by induction. Next, for each i, j, and T such that $1 \leq i \leq m-2$, $1 \leq j \leq n-2$, and $T \subseteq \{1, 2, \ldots, p-2\}$, we have

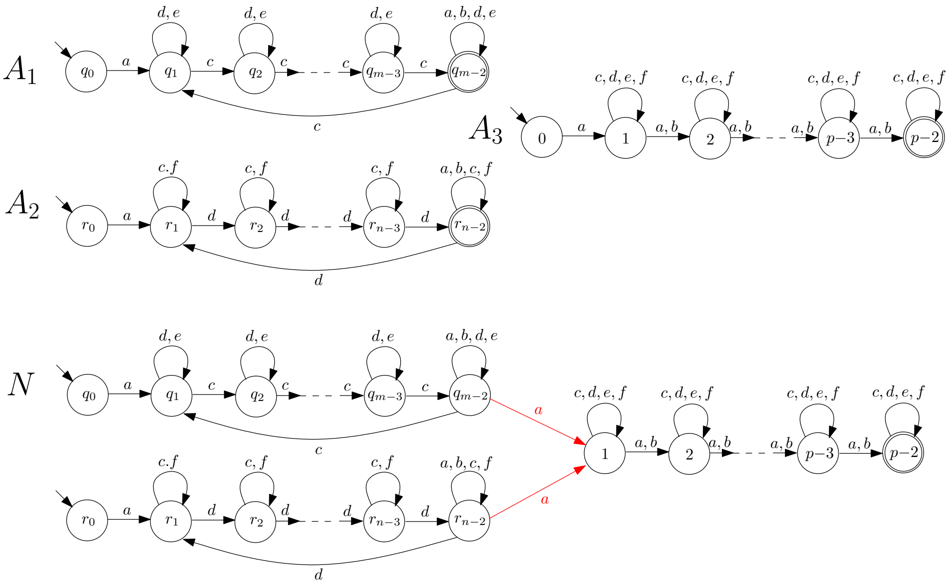

Figure 5: Suffix-free witnesses (top) and NFA N (bottom) for $(L_1 \cup L_2) \cdot L_3$.

- $\{q_{m-2}, r_{n-2}\} \cup T \xrightarrow{c^i d^j} \{q_i, r_j\} \cup T$;
- $\{q_i, r_j\} \cup T \xrightarrow{e} \{q_i\} \cup T$;
- $\{q_i, r_j\} \cup T \xrightarrow{f} \{r_j\} \cup T$; and $\{q_i\} \cup T \xrightarrow{f} T$.

This proves the reachability of $(m-1)(n-1)2^{p-2} + 1$ subsets.

To prove distinguishability, notice that the states $p-2$, q_{m-2}, and r_{n-2} are uniquely distinguishable in NFA N: state $p-2$ is a unique final state of N, the string ea^{p-2} is accepted by N from and only from q_{m-2}, and the string fa^{p-2} is accepted by N from and only from r_{n-2}. Next, consider the subgraph $G(N)$ of unique in-transitions given by transitions (q_0, a, q_1), (q_i, c, q_{i+1}) with $1 \le i \le m-3$, (r_0, a, r_1), (r_j, d, r_{j+1}) with $1 \le j \le n-3$, and $(k, a, k+1)$ where $1 \le k \le p-3$; see dashed transitions in Figure 5 (bottom). This subgraph consists of three paths ending in states q_{m-2}, r_{n-2}, and $p-2$, respectively. Moreover, each state of N is on one of these three paths. Hence from each state of N a uniquely distinguishable state is reached in $G(N)$. By Proposition 2.4, the subset automaton of N does not have equivalent states. □

5. State Complexity of $(L_1 \cap L_2) \cdot L_3$ and $L_1 \cdot (L_2 \cap L_3)$

We consider the state complexity of $(L_1 \cap L_2) \cdot L_3$ for prefix-free regular languages L_1, L_2, and L_3. We get the tight upper bound $(m-2)(n-2)+p$. Our worst-case examples are defined over a binary alphabet. Then we consider the same operation for suffix-free languages. We get an upper bound $((m-2)(n-2)+1)2^{p-2}+1$, and prove its tightness using a quaternary alphabet. Notice that in both cases, no saving is obtained with respect to the composition of the operations.

Theorem 5.1 *Let $m, n, p \ge 3$, and L_1, L_2, L_3 be regular prefix-free languages over an alphabet Σ with $\mathcal{SC}(L_1) = m$, $\mathcal{SC}(L_2) = n$, and $\mathcal{SC}(L_3) = p$. Then $\mathcal{SC}((L_1 \cap L_2) \cdot L_3) \le (m-2)(n-2)+p$, and the bound is tight if $|\Sigma| \ge 2$.*

Proof. We compute the upper bound by the composition of the state complexity of intersection and catenation for prefix-free regular languages. For prefix-free regular languages, the state complexity of intersection is $mn-2(m+n)+6$, and the state complexity of catenation is $m+n-2$ [9]. Thus, the upper bound for $(L_1 \cap L_2) \cdot L_3$ is $mn-2(m+n)+6+p-2 = (m-2)(n-2)+p$. To prove tightness, consider the binary prefix-free languages accepted by the DFAs A_1, A_2, A_3 shown in Figure 6 (top). Notice that A_1 and A_2 are binary witnesses for intersection on prefix-free languages [13, Theorem 1]. □

Theorem 5.2 *Let $m, n, p \geq 4$, and L_1, L_2, L_3 be suffix-free languages over Σ with $\mathcal{SC}(L_1) = m$, $\mathcal{SC}(L_2) = n$, and $\mathcal{SC}(L_3) = p$. Then $\mathcal{SC}((L_1 \cap L_2) \cdot L_3) \leq ((m-2)(n-2)+1)2^{p-2}+1$, and the bound is tight if $|\Sigma| \geq 4$.*

Proof. We compute the upper bound by the composition of the state complexity of intersection and catenation for suffix-free regular languages. For suffix-free regular languages, the state complexity of intersection is $(m-2)(n-2)+2$ and the state complexity of catenation is $(m-1)2^{n-2}+1$ [9]. Thus, the upper bound for $(L_1 \cap L_2) \cdot L_3$ is $((m-2)(n-2)+1)2^{p-2}+1$. To prove tightness, consider the binary prefix-free languages accepted by the DFAs A_1, A_2, A_3 shown in Figure 7 (top). Notice that A_1 and A_2 are binary witnesses for intersection on suffix-free languages [14, Lemma 6]. □

Now we consider the state complexity of $L_1 \cdot (L_2 \cap L_3)$ for prefix-free regular languages L_1, L_2 and L_3. We get an upper bound as the composition of state complexities of catenation and intersection for prefix-free languages. Then we describe prefix-free languages over a binary alphabet meeting this upper bound.

Theorem 5.3 *Let $m, n, p \geq 3$, and L_1, L_2, L_3 be regular prefix-free languages over an alphabet Σ with $\mathcal{SC}(L_1) = m$, $\mathcal{SC}(L_2) = n$, and $\mathcal{SC}(L_3) = p$. Then $\mathcal{SC}(L_1 \cdot (L_2 \cap L_3)) \leq m+(n-2)(p-2)$, and the bound is tight if $|\Sigma| \geq 2$.*

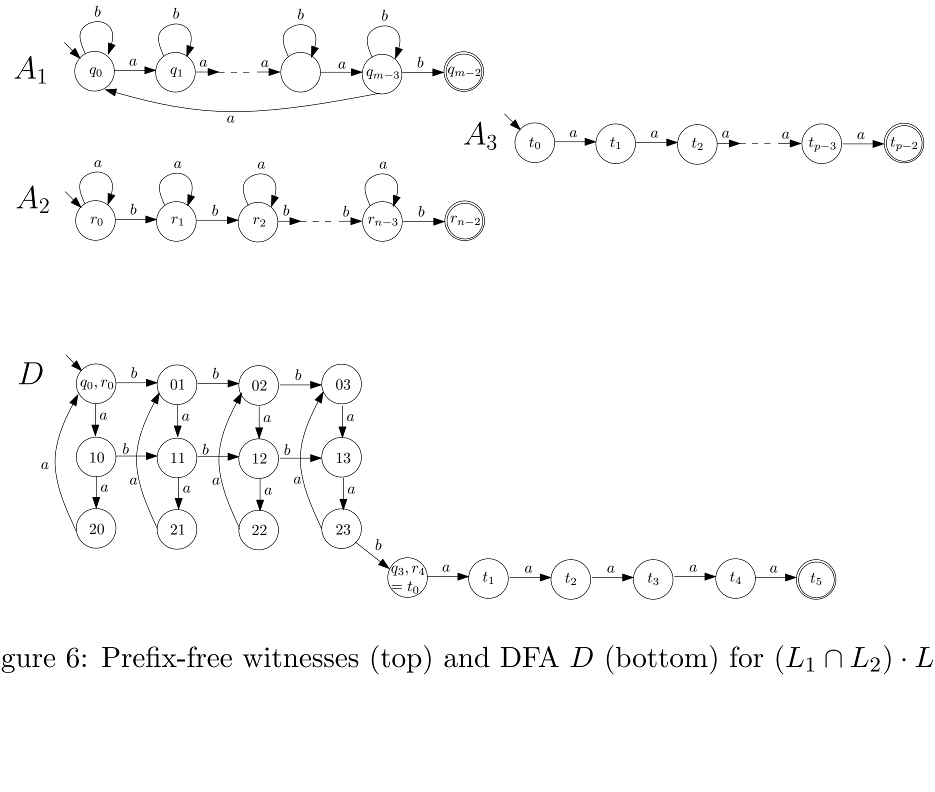

Figure 6: Prefix-free witnesses (top) and DFA D (bottom) for $(L_1 \cap L_2) \cdot L_3$.

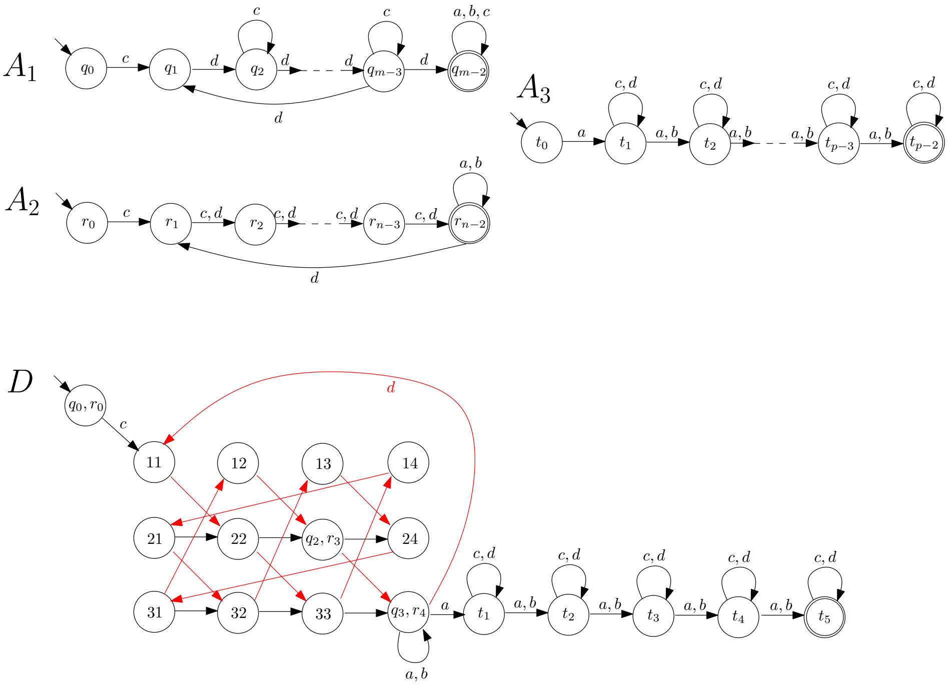

Figure 7: Suffix-free witnesses (top) and NFA N (bottom) for $(L_1 \cap L_2) \cdot L_3$.

Proof. We compute the upper bound by the composition of the state complexity of intersection and catenation for prefix-free regular languages. For prefix-free regular languages, the state complexity of catenation is $m+n-2$, and the state complexity of intersection is $(m-2)(n-2)+2$ [9]. Thus, the upper bound for $L_1 \cdot (L_2 \cap L_3)$ is $m + (n-2)(p-2)$. This gives the upper bound. For tightness, let $L_1 = \{b^{m-2}\}$ and L_2, L_3 be binary prefix-free witnesses for intersection [13, Theorem 1]; see Figure 8 (top). □

We next consider the state complexity of $L_1 \cdot (L_2 \cap L_3)$ for suffix-free regular languages L_1, L_2 and L_3. We compute the upper bound by composition of state complexity of catenation and intersection for suffix-free regular languages. For suffix-free regular languages, the state complexity of intersection is $mn - 2(m+n) + 6$ and the state complexity of catenation is $(m-1)2^{n-2} + 1$ [9]. Thus, the upper bound for $L_1 \cdot (L_2 \cap L_3)$ is $(m-1)2^{(n-2)(p-2)} + 1$.

6. State Complexity of $L_1 \cdot L_2^*$ and $L_1^* \cdot L_2$

We consider the state complexity of $L_1 \cdot L_2^*$ for prefix-free regular languages L_1 and L_2. Let us first recall the construction of a DFA for L_2^*. Let a prefix-free language L_2 be accepted by an n-state DFA $A_2 = (Q_2, \Sigma, \delta_2, s_2, \{f_2\})$ with the sink state d_2. We can construct an n-state DFA for L_2^* from A_2 by making the state f_2 initial, and by replacing each transition (f_2, a, d_2) with the transition $(f_2, a, \delta_2(s_2, a))$. We use this construction to get the next result.

Theorem 6.1 *Let $m, n \geq 3$, and L_1, L_2 be regular prefix-free languages over an alphabet Σ with $\mathcal{SC}(L_1) = m$, $\mathcal{SC}(L_2) = n$. Then $\mathcal{SC}(L_1 \cdot L_2^*) \leq m + n - 2$, and the bound is tight if $|\Sigma| \geq 2$.*

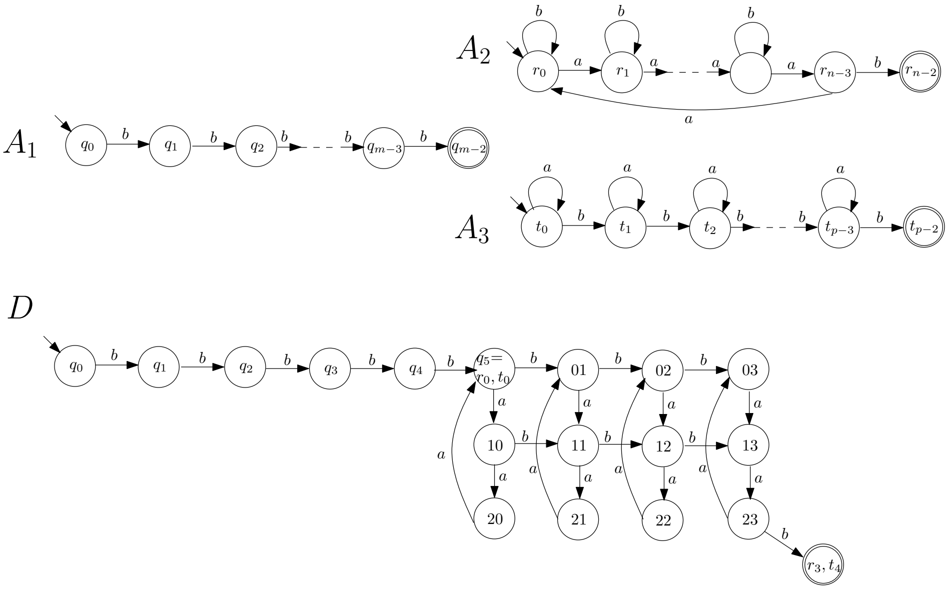

Figure 8: Prefix-free witnesses (top) and DFA D (bottom) for $L_1 \cdot (L_2 \cap L_3)$.

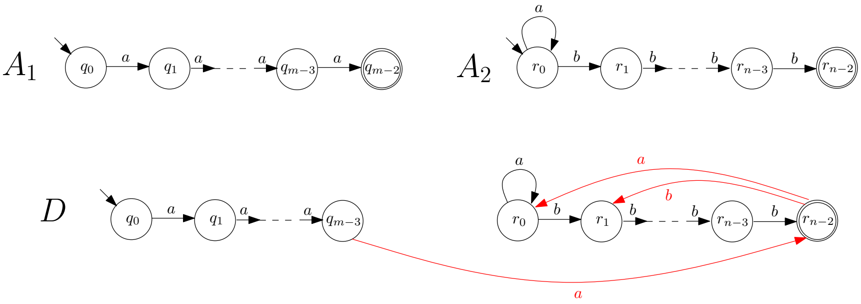

Figure 9: Prefix-free witnesses (top) and DFA D (bottom) for $L_1 \cdot L_2^*$.

Proof. Let $A_1 = (Q_1, \Sigma, \delta_1, s_1, \{f_1\})$ and $A_2 = (Q_2, \Sigma, \delta_2, s_2, \{f_2\})$ be minimal DFAs for prefix-free languages L_1 and L_2, respectively, with sink states d_1 and d_2. Construct an incomplete DFA D for $L_1 \cdot L_2^*$ from DFAs A_1 and A_2 as follows:

(1) omit the states f_1, d_1, d_2 and all the transitions going to or from these states;
(2) for each symbol a add the transition $(f_2, a, \delta_2(s_2, a))$;
(3) for each transition (q, a, f_1) in A_1, add the transition (q, a, f_2);
(4) the initial state is s_1 and the final state is f_2.

By adding the sink state, we get a DFA for $L_1 \cdot L_2^*$ of $m + n - 2$ states, which proves the upper bound. For tightness, consider binary prefix-free languages L_1 and L_2 accepted by DFAs shown in Figure 9 (top). □

Now we consider the state complexity of $L_1 \cdot L_2^*$ for suffix-free regular languages L_1 and L_2. Since the empty string is in L_2^*, we have $L_1 \subseteq L_1 \cdot L_2^*$. Notice that the upper bound coincide with the one for the catenation of suffix-free languages.

Theorem 6.2 *Let $m, n \geq 4$, and L_1, L_2 be suffix-free languages over Σ with $\mathcal{SC}(L_1) = m$ and $\mathcal{SC}(L_2) = n$. Then $\mathcal{SC}(L_1 \cdot L_2^*) \leq (m-1)2^{n-2} + 1$, and the bound is tight if $|\Sigma| \geq 4$.*

Proof. Let L_1, L_2 be suffix-free languages accepted by minimal DFAs $A_1 = (Q_1, \Sigma, \delta_1, s_1, F_1)$ and $A_2 = (Q_2, \Sigma, \delta_2, s_2, F_2)$, with sink states d_1, d_2, respectively. Then A_1, A_2 are non-returning. Construct an NFA N for $L_1 \cdot L_2^*$ from DFAs A_1, A_2 as follows:

(1) omit the states d_1, s_2, d_2 and all the transitions going to or from these states;
(2) for each r in F_2 and each a in Σ, add the transition $(r, a, \delta_2(s_2, a))$;
(3) for each q in F_1 and each a in Σ, add the transition $(q, a, \delta_2(s_2, a))$;
(4) the initial state of N is s_1, and the set of final states is $F_1 \cup F_2$;

see Figure 10 for an example. Since A_1, A_2 are non-returning DFAs, in the subset automaton of N, only the following states may be reachable:

- $\{s_1\}$;
- $\{q\} \cup R$ and R, where $q \in Q_1 \setminus \{s_1, d_1\}$ and $R \subseteq Q_2 \setminus \{s_2, d_2\}$.

In total, we get at most $(m-1)2^{n-2}+1$ reachable states. For tightness, consider the languages L_1 and L_2 accepted by DFAs A_1 and A_2 shown in Figure 10 (top). □

We conclude the paper with the state complexity of $L_1^* \cdot L_2$ on prefix-free and suffix-free languages. In both cases, we get tight upper bounds. Our worst-case examples are defined over a growing alphabet of size $n+3$ for prefix-free languages, and over a 5-letter alphabet for suffix-free languages.

Theorem 6.3 *Let $m, n \geq 4$, and L_1, L_2 be prefix-free languages over Σ with $\mathcal{SC}(L_1) = m$ and $\mathcal{SC}(L_2) = n$. Then $\mathcal{SC}(L_1^* \cdot L_2) \leq (m-1)(2^{n-1}-1)+1$, and the bound is tight if $|\Sigma| \geq n+3$.*

Proof. Let $A_1 = (Q_1, \Sigma, \delta_1, s_1, \{f_1\})$ and $A_2 = (Q_2, \Sigma, \delta_2, s_2, \{f_2\})$ be minimal DFAs for prefix-free languages L_1 and L_2, respectively, with sink states d_1 and d_2. Construct an NFA N for $L_1^* \cdot L_2$ from DFAs A_1 and A_2 as follows:

(1) omit the states d_1, d_2 and all the transitions going to or from these states;
(2) for each symbol a add the transition $(f_1, a, \delta_1(s_1, a))$; denote the resulting DFA by A_1^*
(3) for each transition (q, a, f_1) in A_1^*, add the transition (q, a, s_2);
(4) the set of initial states of N is $\{f_1, s_2\}$ and the final state is f_2.

In the subset automaton of N, only the following sets can be reachable and pairwise distinguishable:

- $\{f_1\} \cup R$, where $R \subseteq Q_2 \setminus \{d_2\}$ and $s_2 \in R$;
- $\{q\} \cup R$ and R, where $q \in Q_1 \setminus \{f_1, d_1\}$ and $R \subsetneq Q_2 \setminus \{d_2\}$;

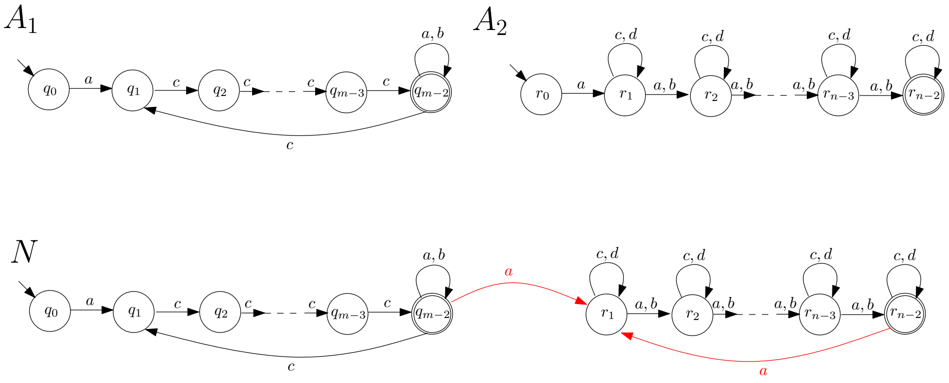

Figure 10: Suffix-free witnesses (top) and NFA N (bottom) for $L_1 \cdot L_2^*$.

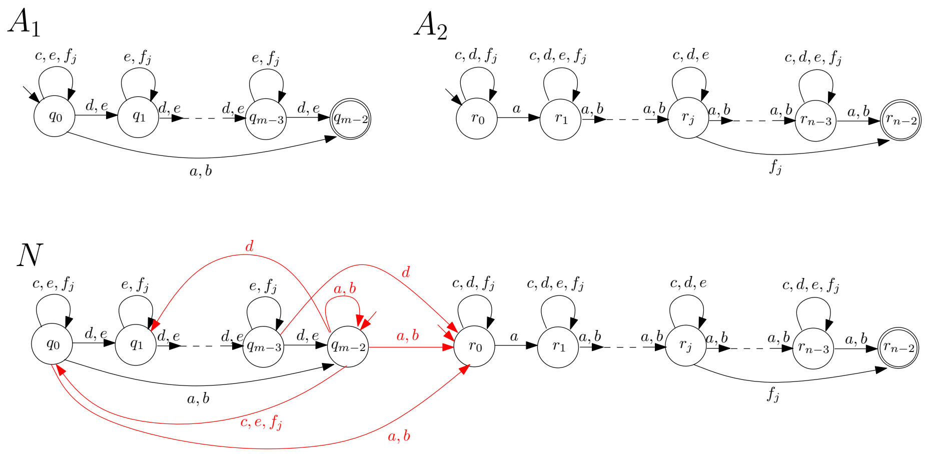

Figure 11: Prefix-free witnesses (top) and NFA N (bottom) for $L_1^* \cdot L_2$.

notice that $\{q\} \cup Q_2 \setminus \{d_2\}$ cannot be reachable if $q \neq f_1$, and each set $\{s_1\} \cup R$ is equivalent to $\{f_1\} \cup R$ since the states s_1 and f_1 go to the same sets on each symbol in N. It follows that the subset automaton of N has at most $1 + (m-2)(2^{n-1}-1) + (2^{n-1}-1) = (m-1)(2^{n-1}-1)+1$ reachable and pairwise distinguishable subsets. For tightness, consider prefix-free languages L_1 and L_2 accepted by DFAs shown in Figure 11 (top). □

Theorem 6.4 *Let $m, n \geq 4$, and L_1, L_2 be suffix-free languages over Σ with $\mathcal{SC}(L_1) = m$ and $\mathcal{SC}(L_2) = n$. Then $\mathcal{SC}(L_1^* \cdot L_2) \leq 2^{m+n-4} + 1$, and the bound is tight if $|\Sigma| \geq 5$.*

Proof. Let L_1, L_2 be suffix-free languages accepted by minimal DFAs $A_1 = (Q_1, \Sigma, \delta_1, s_1, F_1)$ and $A_2 = (Q_2, \Sigma, \delta_2, s_2, F_2)$, with sink states d_1, d_2, respectively. Then A_1, A_2 are non-returning. Construct an NFA N for $L_1^* \cdot L_2$ from DFAs A_1, A_2 as follows:
(1) omit the states d_1, s_2, d_2 and all the transitions going to or from these states;
(2) for each a in Σ and each q in F_1, if $\delta_1(s_1, a) \neq d_1$, then add the transition $(q, a, \delta_1(s_1, a))$;
(3) for each a in Σ and each q in $F_1 \cup \{s_1\}$, if $\delta_2(s_2, a) \neq d_2$, then add $(q, a, \delta_2(s_2, a))$;
(4) the initial state of N is s_1, and the set of final states is F_2;
see Figure 12 for an example. The resulting NFA is non-returning and has $(m+n-4)+1$ states. The corresponding subset automaton has at most $2^{m+n-4}+1$ reachable states which gives the upper bound. To prove tightness, consider the languages L_1 and L_2 accepted by DFAs A_1 and A_2 shown in Figure 12 (top). □

7. Conclusions

We can usually obtain a much lower state complexity for combined operations compared with the compositions of state complexities of individual operations. However, for some cases, the state complexity of combined operations and the composition of state complexities are the same. We have examined prefix-free and suffix-free regular languages and computed the state complexity of combined operations. Table 1 summarizes our results. It also displays the size of alphabet used for describing our worst-case examples.

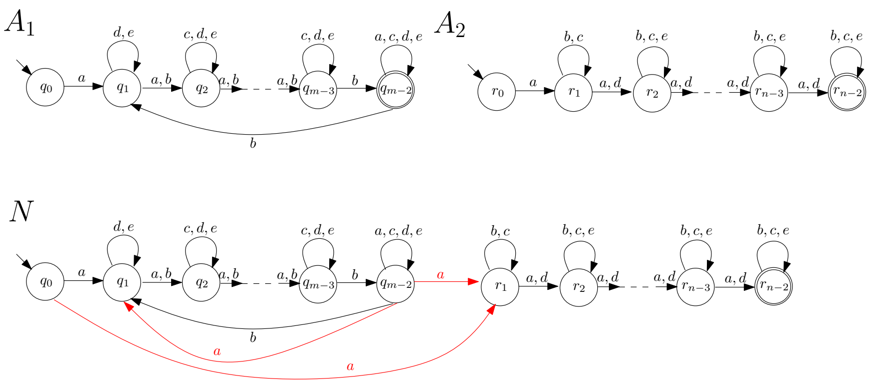

Figure 12: Suffix-free witnesses (top) and NFA N (bottom) for $L_1^* \cdot L_2$.

operation	prefix-free	$\lvert\Sigma\rvert$	suffix-free	$\lvert\Sigma\rvert$
$L_1 \cdot (L_2 \cup L_3)$	$m + np - 4$	3	$(m-1)2^{n+p-4} + 1$	6
$(L_1 \cup L_2) \cdot L_3$	$(m-2)(n-2) + (m+n-4)p + (p^2 - p + 2)/2$	2	$(m-1)(n-1)2^{p-2} + 1$	6
$(L_1 \cap L_2) \cdot L_3$	$(m-2)(n-2) + p$	2	$((m-2)(n-2)+1)2^{p-2} + 1$	4
$L_1 \cdot (L_2 \cap L_3)$	$m + np - 2(n+p) + 4$	2	$\leq (m-1)2^{(n-2)(p-2)} + 1$	-
$L_1 \cdot L_2^*$	$m + n - 2$	2	$(m-1)2^{n-2} + 1$	4
$L_1^* \cdot L_2$	$(m-1)(2^{n-1} - 1) + 1$	$n+3$	$2^{m+n-4} + 1$	5

Table 1: State complexity of combined operations on prefix-free and suffix-free languages; $m, n \geq 4$.

References

[1] J. A. BRZOZOWSKI, G. JIRÁSKOVÁ, B. LIU, A. RAJASEKARAN, M. SZYKULA, On the State Complexity of the Shuffle of Regular Languages. In: C. CÂMPEANU, F. MANEA, J. SHALLIT (eds.), *Descriptional Complexity of Formal Systems - 18th IFIP WG 1.2 International Conference, DCFS 2016, Bucharest, Romania, July 5-8, 2016. Proceedings*. Lecture Notes in Computer Science 9777, Springer, 2016, 73–86.
`http://dx.doi.org/10.1007/978-3-319-41114-9_6`

[2] C. CÂMPEANU, K. CULIK II, K. SALOMAA, S. YU, State Complexity of Basic Operations on Finite Languages. In: *Proceedings of WIA'99*. Lecture Notes in Computer Science 2214, 2001, 60–70.

[3] C. CÂMPEANU, K. SALOMAA, S. YU, Tight Lower Bound for the State Complexity of Shuffle of Regular Languages. *Journal of Automata, Languages and Combinatorics* 7 (2002) 3, 303–310.

[4] R. CMORIK, G. JIRÁSKOVÁ, Basic Operations on Binary Suffix-Free Languages. In: *Proceeding of MEMICS'11*. Lecture Notes in Computer Science 7119, 2011, 94–102.

[5] M. DOMARATZKI, State Complexity of Proportional Removals. *Journal of Automata, Languages and Combinatorics* 7 (2002) 4, 455–468.

[6] M. DOMARATZKI, K. SALOMAA, State complexity of shuffle on trajectories. *Journal of Automata, Languages and Combinatorics* 9 (2004) 2-3, 217–232.

[7] Y.-S. HAN, K. SALOMAA, State Complexity of Union and Intersection of Finite Languages. *International Journal of Foundations of Computer Science* 19 (2008) 3, 581–595.

[8] Y.-S. HAN, K. SALOMAA, State Complexity of Basic Operations on Suffix-Free Regular Languages. *Theoretical Computer Science* 410 (2009) 27-29, 2537–2548.

[9] Y.-S. HAN, K. SALOMAA, D. WOOD, Operational State Complexity of Prefix-Free Regular Languages. In: *Automata, Formal Languages, and Related Topics - Dedicated to Ferenc Gécseg on the occasion of his 70th birthday*. 2009, 99–115.

[10] M. HOLZER, M. KUTRIB, Nondeterministic Descriptional Complexity Of Regular Languages. *International Journal of Foundations of Computer Science* 14 (2003) 6, 1087–1102.

[11] M. HRICKO, G. JIRÁSKOVÁ, A. SZABARI, Union and Intersection of Regular Languages and Descriptional Complexity. In: *Proceedings of DCFS'05*. 2005, 170–181.

[12] J. JIRÁSEK, G. JIRÁSKOVÁ, A. SZABARI, State complexity of concatenation and complementation. *International Journal of Foundations of Computer Science* 16 (2005) 3, 511–529.

[13] G. JIRÁSKOVÁ, M. KRAUSOVÁ, Complexity in Prefix-Free Regular Languages. In: I. MCQUILLAN, G. PIGHIZZINI (eds.), *Proceedings Twelfth Annual Workshop on Descriptional Complexity of Formal Systems, DCFS 2010, Saskatoon, Canada, 8-10th August 2010.*. EPTCS 31, 2010, 197–204.

[14] G. JIRÁSKOVÁ, P. OLEJÁR, State Complexity of Intersection and Union of Suffix-Free Languages and Descriptional Complexity. In: H. BORDIHN, R. FREUND, M. HOLZER, M. KUTRIB, F. OTTO (eds.), *Workshop on Non-Classical Models for Automata and Applications - NCMA 2009, Wroclaw, Poland, August 31 - September 1, 2009. Proceedings*. books@ocg.at 256, Austrian Computer Society, 2009, 151–166.

[15] A. MASLOV, Estimates of the number of states of finite automata. *Soviet Mathematics Doklady* 11 (1970), 1373–1375.

[16] C. NICAUD, Average State Complexity of Operations on Unary Automata. In: *Proceedings of MFCS'99*. Lecture Notes in Computer Science 1672, 1999, 231–240.

[17] G. PIGHIZZINI, J. SHALLIT, Unary Language Operations, State Complexity and Jacobsthal's Function. *International Journal of Foundations of Computer Science* 13 (2002) 1, 145–159.

[18] A. SALOMAA, D. WOOD, S. YU, On the state complexity of reversals of regular languages. *Theoretical Computer Science* 320 (2004) 2-3, 315–329.

[19] D. WOOD, *Theory of Computation*. John Wiley & Sons, Inc., New York, NY, 1987.

[20] S. YU, State Complexity of Regular Languages. *Journal of Automata, Languages and Combinatorics* 6 (2001) 2, 221–234.

[21] S. YU, Q. ZHUANG, K. SALOMAA, The state complexities of some basic operations on regular languages. *Theoretical Computer Science* 125 (1994) 2, 315–328.