State Complexity of Single-Word Pattern Matching in Regular Languages***

Janusz A. Brzozowski[1], Sylvie Davies[2], and Abhishek Madan[1]

[1] David R. Cheriton School of Computer Science, University of Waterloo, Waterloo, ON, Canada N2L 3G1
brzozo@uwaterloo.ca, a7madan@edu.uwaterloo.ca
[2] Department of Pure Mathematics, University of Waterloo Waterloo, ON, Canada N2L 3G1
sldavies@uwaterloo.ca

Abstract. The state complexity $\kappa(L)$ of a regular language L is the number of states in the minimal deterministic finite automaton recognizing L. In a general pattern-matching problem one has a set T of texts and a set P of patterns; both T and P are sets of words over a finite alphabet Σ. The matching problem is to determine whether any of the patterns appear in any of the texts, as prefixes, or suffixes, or factors, or subsequences. In previous work we examined the state complexity of these problems when both T and P are regular languages, that is, we computed the state complexity of the languages $(P\Sigma^*) \cap T$, $(\Sigma^* P) \cap T$, $(\Sigma^* P \Sigma^*) \cap T$, and $(\Sigma^* \sqcup P) \cap T$, where $\sqcup$ is the shuffle operation. It turns out that the state complexities of these languages match the naïve upper bounds derived by composing the state complexities of the basic operations used in each expression. However, when P is a single word w, and Σ has two or more letters, the bounds are drastically reduced to the following: $\kappa((w\Sigma^*) \cap T) \leqslant m+n-1$; $\kappa((\Sigma^* w) \cap T) \leqslant (m-1)n-(m-2)$; $\kappa((\Sigma^* w \Sigma^*) \cap T) \leqslant (m-1)n$; and $\kappa((\Sigma^* \sqcup w) \cap T) \leqslant (m-1)n$. The bounds for factor and subsequence matching are the same as the naïve bounds, but this is not the case for prefix and suffix matching. For unary languages, we have a tight upper bound of $m+n-2$ in all four cases.

Keywords: all-sided ideal, combined operation, factor, finite automaton, left ideal, pattern matching, prefix, regular language, right ideal, state complexity, subsequence, suffix, two-sided ideal

1 Introduction

The *state complexity* of a regular language L, denoted $\kappa(L)$, is the number of states in the minimal deterministic finite automaton (DFA) recognizing L. The

* This work was supported by the Natural Sciences and Engineering Research Council of Canada grant No. OGP0000871.

** This is a post-peer-review, pre-copyedit version of an article published in Descriptional Complexity of Formal Systems. The final authenticated version is available online at: https://doi.org/10.1007/978-3-030-23247-4_6

state complexity of an *operation* on regular languages is the worst-case state complexity of the resulting language, expressed in terms of the the input languages' state complexities. A language attaining this worst-case state complexity is called a *witness* for the operation.

The state complexities of "basic" regular operations such as intersection and concatenation have been thoroughly studied [7,8,9]. There has also been some attention devoted towards "combined" operations such as concatenation with Σ^* to form languages called *ideals* [3]. A practical application of ideals is in *pattern matching*, or finding occurrences of a pattern in a text, commonly as either prefixes, suffixes, factors, or subsequences. (For a detailed treatment of pattern matching, see [4].) Brzozowski et al. [1] formulated several pattern matching problems as the construction of a regular language, using the intersection between a text language T and an ideal of a pattern language P. In the general case, given that $\kappa(T) \leqslant n$ and $\kappa(P) \leqslant m$, and denoting $\sqcup\!\sqcup$ as the shuffle operation, the following state complexity bounds were shown to be tight:

1. *Prefix:* $\kappa((P\Sigma^*) \cap T) \leqslant mn$.
2. *Suffix:* $\kappa((\Sigma^* P) \cap T) \leqslant 2^{m-1}n$.
3. *Factor:* $\kappa((\Sigma^* P \Sigma^*) \cap T) \leqslant (2^{m-2}+1)n$.
4. *Subsequence:* $\kappa((P \sqcup\!\sqcup \Sigma^*) \cap T) \leqslant (2^{m-2}+1)n$.

These bounds are in fact the naïve bounds derived from composing the state complexity of the intersection between the Σ^*-concatenated pattern language and the text language. However, these bounds are exponential in m, which leads to the following question: to what degree would restricting P lower the bounds? In this paper, we focus on restricting P to be a single word; that is, $P = \{w\}$.

Single-word pattern matching has many practical applications. For example, a common use of the `grep` utility in Unix is to search for the files in a directory in which a search word appears. In bioinformatics, a DNA sequence t is often searched to locate a sequence of nucleotides w [5]. There has also been work in distributed systems to "learn" common execution patterns from log files and use them to identify anomalous executions in new logs [6].

In this paper, we show that for languages T and $\{w\}$ such that $\kappa(T) \leqslant n$ and $\kappa(\{w\}) \leqslant m$, the following upper bounds hold:

1. Prefix: $\kappa((w\Sigma^*) \cap T) \leqslant m + n - 1$.
2. Suffix: $\kappa((\Sigma^* w) \cap T) \leqslant (m-1)n - (m-2)$.
3. Factor: $\kappa((\Sigma^* w \Sigma^*) \cap T) \leqslant (m-1)n$.
4. Subsequence: $\kappa((\Sigma^* \sqcup\!\sqcup w) \cap T) \leqslant (m-1)n$.

Furthermore, in each case there exist languages T_n and $\{w\}_m$ that meet the upper bounds. All of these bounds can be achieved using a binary alphabet, but not using a unary alphabet.

2 Terminology and Notation

A *deterministic finite automaton (DFA)* is a 5-tuple $\mathcal{D} = (Q, \Sigma, \delta, q_0, F)$, where Q is a finite non-empty set of *states*, Σ is a finite non-empty *alphabet*, $\delta\colon Q \times \Sigma \to$

Q is the *transition function*, $q_0 \in Q$ is the *initial* state, and $F \subseteq Q$ is the set of *final* states. We extend δ to functions $\delta\colon Q \times \Sigma^* \to Q$ and $\delta\colon 2^Q \times \Sigma^* \to 2^Q$ as usual.

A language $L(\mathcal{D})$ is *accepted* by $\mathcal{D}$ if, for all $w \in L(\mathcal{D})$, $\delta(q_0, w) \in F$. If q is a state of $\mathcal{D}$, then the language $L_q(\mathcal{D})$ of q is the language accepted by the DFA $(Q, \Sigma, \delta, q, F)$. Let L be a language over Σ. The *quotient* of L by a word $x \in \Sigma^*$ is the set $x^{-1}L = \{y \in \Sigma^* \mid xy \in L\}$. In a DFA $\mathcal{D} = (Q, \Sigma, \delta, q_0, F)$, if $\delta(q_0, w) = q$, then $L_q(\mathcal{D}) = w^{-1}L(\mathcal{D})$.

Two states p and q of $\mathcal{D}$ are *indistinguishable* if $L_p(\mathcal{D}) = L_q(\mathcal{D})$. A state q is *reachable* if there exists $w \in \Sigma^*$ such that $\delta(q_0, w) = q$. A DFA $\mathcal{D}$ is *minimal* if it has the smallest number of states and the smallest alphabet among all DFAs accepting $L(\mathcal{D})$. It is well known that a DFA is minimal if it uses the smallest alphabet, all of its states are reachable, and no two states are indistinguishable.

We sometimes define transition functions as transformations induced by letters, written as $a\colon t$ where $t\colon Q \to Q$, for all $a \in \Sigma$. In particular, we use $\mathbb{1}$ to denote the identity transformation (i.e., $\delta(q, a) = q$ for all $q \in Q$), and $(q_0, q_1, \ldots, q_{k-1})$ to denote a *k-cycle*, where $\delta(q_i, a) = q_{i+1}$ for $0 \leqslant i \leqslant k-2$ and $\delta(q_{k-1}, a) = q_0$. For states not in $\{q_0, q_1, \ldots, q_{k-1}\}$, the k-cycle acts as the identity transformation.

Throughout the paper, we fix $w = a_1 \cdots a_{m-2}$, where $a_i \in \Sigma$ for $1 \leqslant i \leqslant m-2$. Let $w_0 = \varepsilon$ (where ε denotes the empty word) and for $1 \leqslant i \leqslant m-2$, let $w_i = a_1 \cdots a_i$. We write $W = \{w_0, w_1, \ldots, w_{m-2}\}$ for the set of all prefixes of w. Note that if the state complexity of $\{w\}$ is m, then w is of length $m-2$.

3 Matching a Single Prefix

Theorem 1. *Suppose $m \geqslant 3$ and $n \geqslant 2$. If w is a non-empty word, $\kappa(\{w\}) \leqslant m$ and $\kappa(T) \leqslant n$ then we have*

$$\kappa((w\Sigma^*) \cap T) \leqslant \begin{cases} m+n-1, & \text{if } |\Sigma| \geqslant 2; \\ m+n-2, & \text{if } |\Sigma| = 1. \end{cases}$$

Furthermore, these upper bounds are tight.

Remark 1. When $|\Sigma| = 1$ (that is, P and T are languages over a unary alphabet), the tight upper bound $m+n-2$ actually holds in *all four cases* we consider in this paper. This is because if L is a language over a unary alphabet Σ, then the ideals $L\Sigma^*$, Σ^*L, $\Sigma^*L\Sigma^*$ and $\Sigma^* \sqcup L$ coincide; thus the prefix, suffix, factor and subsequence matching cases coincide.

Proof. We first derive upper bounds for the two cases of $|\Sigma|$.
Upper Bounds: Let $\mathcal{D}_T = (Q, \Sigma, \delta, q_0, F_T)$, where $Q = \{q_0, \ldots, q_{n-1}\}$, be a DFA accepting T. Let $P = \{w\}$ and let the minimal DFA of P be $\mathcal{D}_P = (W \cup \{\emptyset\}, \Sigma, \alpha, w_0, \{w_{m-2}\})$. Here w_{m-2} is the only final state, and $\emptyset$ is the empty state. Define α as follows: for $0 \leqslant i \leqslant m-2$, we set

$$\alpha(w_i, a) = \begin{cases} w_{i+1}, & \text{if } a = a_i; \\ \emptyset, & \text{otherwise.} \end{cases}$$

Also define $\alpha(\emptyset, a) = \emptyset$ for all $a \in \Sigma$. Let the state reached by w in $\mathcal{D}_T$ be $q_r = \delta(q_0, w)$; we construct a DFA $\mathcal{D}_L$ that accepts $L = (w\Sigma^*) \cap T$. As shown in Figure 1, let $\mathcal{D}_L = (Q \cup (W \setminus \{w_{m-2}\}) \cup \{\emptyset\}, \Sigma, \beta, w_0, F_T)$, where β is defined as follows: for $q \in Q \cup (W \setminus \{w_{m-2}\}) \cup \{\emptyset\}$ and $a \in \Sigma$,

$$\beta(q,a) = \begin{cases} \delta(q,a), & \text{if } q \in Q; \\ \alpha(q,a), & \text{if } q \in W \setminus \{w_{m-2}, w_{m-3}\}; \\ q_r, & \text{if } q = w_{m-3}, \text{ and } a = a_{m-2}; \\ \emptyset, & \text{otherwise.} \end{cases}$$

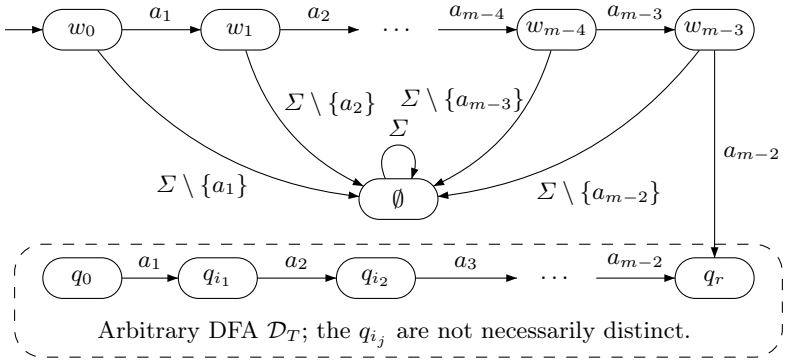

Fig. 1. DFA $\mathcal{D}_L$ for matching a single prefix. The final state set F_T is a subset of the states from the arbitrary DFA $\mathcal{D}_T$; final states are not marked on the diagram.

Recall that in a DFA $\mathcal{D}$, if state q is reached from the initial state by a word u, then the language of q is equal to the quotient of $L(\mathcal{D})$ by u. Thus the language of state q_r is the quotient of T by w, that is, the set $w^{-1}T = \{y \in \Sigma^* \mid wy \in T\}$. The DFA $\mathcal{D}_L$ accepts a word x if and only if it has the form wy for $y \in w^{-1}T$; we need the prefix w to reach the arbitrary DFA $\mathcal{D}_T$, and w must be followed by a word that sends q_r to an accepting state, that is, a word y in the language $w^{-1}T$ of q_r. So $L = \{wy \mid y \in w^{-1}T\} = \{wy \mid y \in \Sigma^*, wy \in T\} = (w\Sigma^*) \cap T$. That is, L is the set of all words of T that begin with w, as required. It follows that the state complexity of L is less than or equal to $m + n - 1$. If $|\Sigma| = 1$, all the $\Sigma \setminus \{a_i\}$ are empty and state $\emptyset$ is not needed. Hence the state complexity of L is less than or equal to $m + n - 2$ in this case.

Lower Bound, $|\Sigma| = 1$: For $m \geqslant 3$, let $P = \{a^{m-2}\}$ where $\kappa(P) = m$. For $n \geqslant 2$, let T be the language of the DFA $\mathcal{D}_n = (Q_n, \{a\}, \delta_1, 0, \{r-1\})$, where $\kappa(T) = n$, δ_1 is defined by $a\colon (0, 1, \ldots, n-1)$, and $r = \delta_1(0, a^{m-2})$. Let $\mathcal{D}_L$ be the DFA shown in Figure 2 for the language $L = (P\Sigma^*) \cap T$. Obviously $\mathcal{D}_L$ has $m + n - 2$ states and they are all reachable. Since the shortest word accepted from any state is distinct from that of any other state, all the states

are pairwise distinguishable. Hence P and T constitute witnesses that meet the required bound.

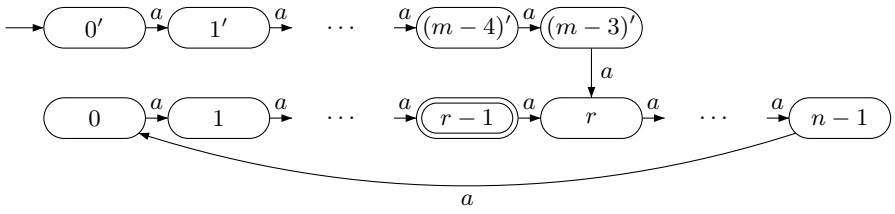

Fig. 2. Minimal DFA of L for the case $|\Sigma| = 1$.

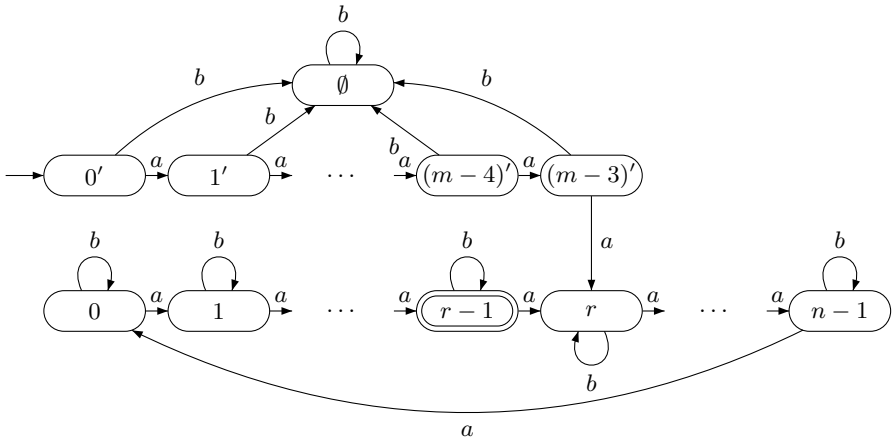

Fig. 3. Minimal DFA of L for the prefix case with $|\Sigma| > 1$.

Lower Bound, $|\Sigma| \geqslant 2$: For $m \geqslant 3$, let $P = \{a^{m-2}\}$ where $\kappa(P) = m$. For $n \geqslant 2$, let T be the language of the DFA $\mathcal{D}_n = (Q_n, \{a, b\}, \delta_2, 0, \{r-1\})$ where $\kappa(T) = n$, δ_2 is defined by $a\colon (0, 1, \ldots, n-1)$ and $b\colon \mathbb{1}$, and $r = \delta_2(0, a^{m-2})$. Construct the DFA $\mathcal{D}_L$ for the language $L = (P\Sigma^*) \cap T$ as is shown in Figure 3. It is clear that all states are reachable and distinguishable by their shortest accepted words. □

4 Matching a Single Suffix

Let $w, x, y, z \in \Sigma^*$. We introduce some notation:

- $x \preceq_p y$ means x is a prefix of y, and $x \succeq_s y$ means x has y as a suffix.
- If $x \succeq_s y$ and $y \preceq_p z$, we say y is a *bridge* from x to z or that y *connects* x to z. We also denote this by $x \to y \to z$.

– $x \twoheadrightarrow y \twoheadrightarrow z$ means that y is the *longest* bridge from x to z. That is, $x \to y \to z$, and whenever $x \to w \to z$ we have $|w| \leqslant |y|$. Equivalently, y is the longest suffix of x that is also a prefix of z.

Proposition 1. *If the state complexity of $\{w\}$ is m, then the state complexity of $\Sigma^* w$ is $m-1$.*

Proof. Let $\mathcal{A} = (W, \Sigma, \delta_{\mathcal{A}}, w_0, \{w_{m-2}\})$ be the DFA with transitions defined as follows: for all $a \in \Sigma$ and $w_i \in W$, we have $w_i a \twoheadrightarrow \delta_{\mathcal{A}}(w_i, a) \twoheadrightarrow w$. That is, $\delta_{\mathcal{A}}(w_i, a)$ is defined to be the maximal-length bridge from $w_i a$ to w, or equivalently, the longest suffix of $w_i a$ that is also a prefix of w. Note that if $a = a_{i+1}$, then $\delta_{\mathcal{A}}(w_i, a) = w_{i+1}$.

We observe that every state $w_i \in W$ is reachable from w_0 by the word w_i, and that each state w_i is distinguished from all other states by $a_{i+1} \cdots a_{m-2}$. It remains to be shown that $\Sigma^* w = L(\mathcal{A})$. In the following, for convenience, we simply write δ rather than $\delta_{\mathcal{A}}$.

We claim that for $x \in \Sigma^*$, we have $w_i x \twoheadrightarrow \delta(w_i, x) \twoheadrightarrow w$. That is, the defining property of the transition function extends nicely to words. Recall that the extension of δ to words is defined inductively in terms of the behavior of δ on letters, so it is not immediately clear that this property carries over to words.

We prove this claim by induction on $|x|$. If $x = \varepsilon$, this is clear. Now suppose $x = ya$ for some $y \in \Sigma^*$ and $a \in \Sigma$, and that $w_i y \twoheadrightarrow \delta(w_i, y) \twoheadrightarrow w$. Let $\delta(w_i, y) = w_j$ and let $\delta(w_i, x) = \delta(w_j, a) = w_k$. We want to show that $w_i x \twoheadrightarrow w_k \twoheadrightarrow w$.

First we show that $w_i x \to w_k \to w$. We know $w_k \preceq_p w$, so it remains to show that $w_i x \succeq_s w_k$. Since $w_k = \delta(w_i, x) = \delta(w_j, a)$, by definition we have $w_j a \twoheadrightarrow w_k \twoheadrightarrow w$. Since $\delta(w_i, y) = w_j$, we have $w_i y \twoheadrightarrow w_j \twoheadrightarrow w$. In particular, $w_i y \succeq_s w_j$ and thus $w_i x = w_i y a \succeq_s w_j a$. Thus $w_i x \succeq_s w_j a \succeq_s w_k$ as required.

Next, we show that whenever $w_i x \to w_\ell \to w$, we have $|w_\ell| \leqslant |w_k|$. If $w_\ell = \varepsilon$, this is immediate, so suppose $w_\ell \neq \varepsilon$. Since $w_i x = w_i y a \succeq_s w_\ell$, and w_ℓ is nonempty, it follow that w_ℓ ends with a. Thus $w_\ell = w_{\ell-1} a$. Since $w_i y a \succeq_s w_{\ell-1} a$, we have $w_i y \succeq_s w_{\ell-1}$. Additionally, $w_{\ell-1} \preceq_p w$, so $w_i y \to w_{\ell-1} \to w$. Since $w_i y \twoheadrightarrow w_j \twoheadrightarrow w$, we have $|w_{\ell-1}| \leqslant |w_j|$. Since $w_i y \succeq_s w_j$ and $w_i y \succeq_s w_{\ell-1}$ and $|w_j| \geqslant |w_{\ell-1}|$, we have $w_j \succeq_s w_{\ell-1}$. Thus $w_j a \succeq_s w_{\ell-1} a = w_\ell$. It follows that $w_j a \to w_\ell \to w$. But recall that $\delta(w_i, x) = \delta(w_j, a) = w_k$, so $w_j a \twoheadrightarrow w_k \twoheadrightarrow w$, and $|w_\ell| \leqslant |w_k|$ as required.

Now, we show that $\mathcal{A}$ accepts the language $\Sigma^* w$. Suppose $x \in \Sigma^* w$ and write $x = yw$. The initial state of $\mathcal{A}$ is $w_0 = \varepsilon$. We have $yw \twoheadrightarrow \delta(\varepsilon, yw) \twoheadrightarrow w$, that is, $\delta(\varepsilon, yw)$ is the longest suffix of yw that is also a prefix of w. But this longest suffix is simply w itself, which is the final state. So x is accepted. Conversely, suppose $x \in \Sigma^*$ is accepted by $\mathcal{A}$. Then $\delta(\varepsilon, x) = w$, and thus $x \twoheadrightarrow w \twoheadrightarrow w$ by definition. In particular, this means $x \succeq_s w$, and so $x \in \Sigma^* w$. □

Next we establish an upper bound on the state complexity of $(\Sigma^* w) \cap T$. The upper bound in this case is quite complicated to derive. Suppose w has state complexity m and T has state complexity at most n, for $m \geqslant 3$ and $n \geqslant 2$. Let $\mathcal{A}$ be the $(m-1)$-state DFA for $\Sigma^* w$ defined in Proposition 1, and let $\mathcal{D}$ be an n-state DFA for T with state set Q_n, transition function α, and final state set

F. The direct product $\mathcal{A} \times \mathcal{D}$ with final state set $\{w\} \times F$ recognizes $(\Sigma^* w) \cap T$. We claim that this direct product has at most $(m-1)n-(m-2)$ reachable and pairwise distinguishable states, and thus the state complexity of $(\Sigma^* w) \cap T$ is at most $(m-1)n-(m-2)$.

Since $\mathcal{A}$ has $m-1$ states and $\mathcal{D}$ has n states, there are at most $(m-1)n$ reachable states. It will suffice to show that for each word w_i with $1 \leqslant i \leqslant m-2$, there exists a word $w_{f(i)} \neq w_i$ and a state $p_i \in Q_n$ such that (w_i, p_i) is indistinguishable from $(w_{f(i)}, p_i)$. This gives $m-2$ states that are each indistinguishable from another state, establishing the upper bound.

We choose $f(i)$ so that $w_i \twoheadrightarrow w_{f(i)} \twoheadrightarrow w_{i-1}$. In other words, $w_{f(i)}$ is the longest suffix of w_i that is also a *proper* prefix of w_i. To find p_i, first observe that there exists a non-final state $q \in Q_n$ and a state $r \in Q_n$ such that $\alpha(r, w) = q$. Indeed, if no such states existed, then for all states r, the state $\alpha(r, w)$ would be final. Thus we would have $\Sigma^* w \subseteq T$, and the state complexity of $(\Sigma^* w) \cap T = \Sigma^* w$ would be $m-1$, which is lower than our upper bound since $n \geqslant 2$. Now, set $p_i = \alpha(r, w_i)$, and note that $\alpha(p_i, a_{i+1}) = p_{i+1}$, and $\alpha(p_i, a_{i+1} \cdots a_{m-2}) = q$.

To establish the upper bound, we will need two technical lemmas. Their proofs can be found in [2].

Lemma 1. *If $i < m-2$ and $a \neq a_{i+1}$, or if $i = m-2$, then $\delta_{\mathcal{A}}(w_i, a) = \delta_{\mathcal{A}}(w_{f(i)}, a)$.*

Lemma 2. *If $i < m-2$, then $\delta_{\mathcal{A}}(w_{f(i)}, a_{i+1}) = w_{f(i+1)}$.*

Proposition 2. *Suppose $m \geqslant 3$ and $n \geqslant 2$. If w is non-empty, $\kappa(\{w\}) \leqslant m$, and $\kappa(T) \leqslant n$, then we have $\kappa((\Sigma^* w) \cap T) \leqslant (m-1)n-(m-2)$.*

Proof. It suffices to prove that states (w_i, p_i) and $(w_{f(i)}, p_i)$ are indistinguishable for $1 \leqslant i \leqslant m-2$. We proceed by induction on the value $m-2-i$.

The base case is $m-2-i = 0$, that is, $i = m-2$. Our states are (w_{m-2}, p_{m-2}) and $(w_{f(m-2)}, p_{m-2})$. By Lemma 1, we have $\delta_{\mathcal{A}}(w_{m-2}, a) = \delta_{\mathcal{A}}(w_{f(m-2)}, a)$ for all $a \in \Sigma$. Thus non-empty words cannot distinguish the states. But recall that $p_{m-2} = q$ is a non-final state, so the states we are trying to distinguish are both non-final, and thus the empty word does not distinguish the states either. So these states are indistinguishable.

Now, suppose $m-2-i > 0$, that is, $i < m-2$. Assume that states (w_{i+1}, p_{i+1}) and $(w_{f(i+1)}, p_{i+1})$ are indistinguishable. We want to show that (w_i, p_i) and $(w_{f(i)}, p_i)$ are indistinguishable. Since $f(i) < i < m-2$, both states are non-final, and thus the empty word cannot distinguish them. By Lemma 1, if $a \neq a_{i+1}$. then $\delta_{\mathcal{A}}(w_i, a) = \delta_{\mathcal{A}}(w_{f(i)}, a)$ for all $a \in \Sigma$. So only words that start with a_{i+1} can possibly distinguish the states. But by Lemma 2, letter a_{i+1} sends the states to (w_{i+1}, p_{i+1}) and $(w_{f(i+1)}, p_{i+1})$, which are indistinguishable by the induction hypothesis. Thus the states cannot be distinguished. □

Next we show that the upper bound of Proposition 2 is tight.

Definition 1. *Let T be the language accepted by the DFA $\mathcal{D}$ with state set Q_n, alphabet Σ, initial state 0, final state set $\{0, \ldots, n-2\}$, and transformations $a\colon (0, \ldots, n-1)$ and $b\colon \mathbb{1}$. See Figure 4.*

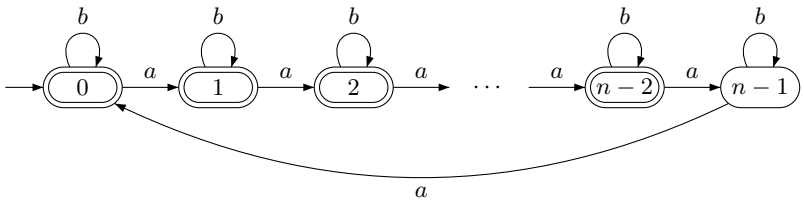

Fig. 4. Witness language T of Definition 1.

Theorem 2. *Suppose $m \geqslant 3$ and $n \geqslant 2$. There exists a word w and a language T, with $\kappa(\{w\}) = m$ and $\kappa(T) = n$, such that $\kappa((\Sigma^* w) \cap T) = (m-1)n-(m-2)$.*

Proof. Let $\Sigma = \{a, b\}$ and let $w = b^{m-2}$. Let $\mathcal{A}$ be the DFA for $\Sigma^* w$. Let T be the language of Definition 1. The DFA $\mathcal{A} \times \mathcal{D}$ is illustrated in Figure 5.

We show that $\mathcal{A} \times \mathcal{D}$ has $(m-1)n-(m-2)$ reachable and pairwise distinguishable states. For reachability, for $0 \leqslant i \leqslant m-2$ and $0 \leqslant q \leqslant n-1$, we can reach (b^i, q) from the initial state $(\varepsilon, 0)$ by the word $a^q b^i$. For distinguishability, note that all $m-1$ states in column $n-1$ are indistinguishable, and so collapse to one state under the indistinguishability relation. Indeed, given states $(b^i, n-1)$ and $(b^j, n-1)$, if we apply a both states are sent to $(\varepsilon, 0)$, and if we apply b we simply reach another pair of non-final states in column $n-1$. Hence at most $(m-1)n-(m-2)$ of the reachable states are pairwise distinguishable. Next consider (b^i, q) and (b^j, q) with $i < j$ and $q \neq n-1$. We can distinguish these states by b^{m-2-j}. So pairs of states in the same column are distinguishable, with the exception of states in column $n-1$. For pairs of states in different columns, consider (b^i, p) and (b^j, q) with $p < q$. If $q \neq n-1$, then by a^{n-1-q} we reach $(\varepsilon, n-1+p-q)$ and $(\varepsilon, n-1)$. These latter states are distinguished by $w = b^{m-2}$. If $q = n-1$, then (b^i, p) and $(b^j, n-1)$ are distinguished by b^{m-2-i}. Hence there are $(m-1)n-(m-2)$ reachable and pairwise distinguishable states. □

5 Matching a Single Factor

Proposition 3. *If the state complexity of $\{w\}$ is m, then the state complexity of $\Sigma^* w \Sigma^*$ is $m-1$.*

Proof. Let $\mathcal{A} = (W, \Sigma, \delta_{\mathcal{A}}, w_0, \{w_{m-2}\})$ be the DFA with transitions defined as follows: for all $a \in \Sigma$ and $w_i \in W$, we have $w_i a \twoheadrightarrow \delta_{\mathcal{A}}(w_i, a) \twoheadrightarrow w$. Recall from Proposition 1 that $\mathcal{A}$ recognizes $\Sigma^* w$. We modify $\mathcal{A}$ to obtain a DFA $\mathcal{A}'$ that accepts $\Sigma^* w \Sigma^*$ as follows.

Let $\mathcal{A}' = (W, \Sigma, \delta_{\mathcal{A}'}, w_0, \{w_{m-2}\})$, where $\delta_{\mathcal{A}'}$ is defined as follows for each $a \in \Sigma$: $\delta_{\mathcal{A}'}(w_i, a) = \delta_{\mathcal{A}}(w_i, a)$ for $i < m-2$, and $\delta_{\mathcal{A}'}(w_{m-2}, a) = w_{m-2}$. Note that $\mathcal{A}'$ is minimal: state w_i can be reached by the word w_i, and states w_i and w_j with $i < j$ are distinguished by $a_{j+1} \cdots a_{m-2}$. It remains to show that $\mathcal{A}'$ accepts $\Sigma^* w \Sigma^*$.

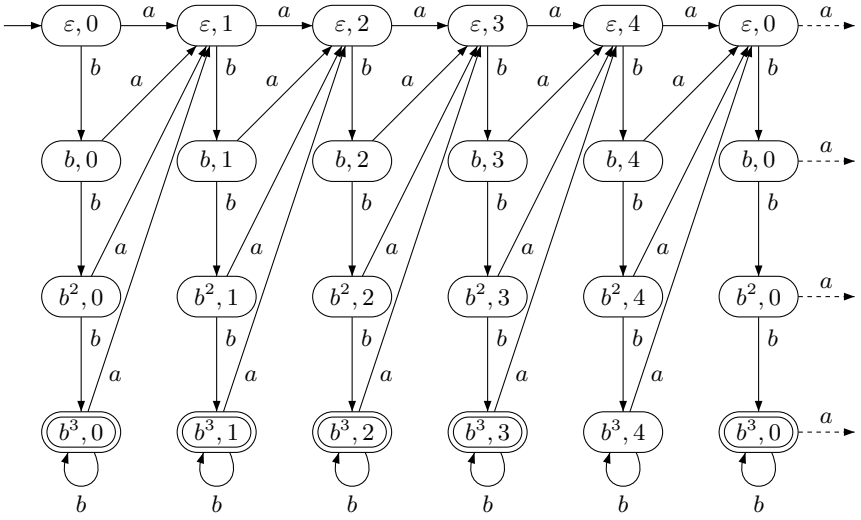

Fig. 5. DFA $\mathcal{A} \times \mathcal{D}$ for matching a single suffix, with $m = 5$ and $n = 5$. Column 0 is duplicated for a cleaner diagram; the DFA contains only one copy of this column.

To simplify the notation, we write δ' instead of $\delta_{\mathcal{A}'}$ and δ instead of $\delta_{\mathcal{A}}$. Suppose x is accepted by $\mathcal{A}'$. Write $x = yz$, where y is the shortest prefix of x such that $\delta'(\varepsilon, y) = w_{m-2}$. Since y is minimal in length, for every proper prefix y' of y, we have $\delta'(\varepsilon, y') = w_i$ for some $i < m - 2$. It follows that $\delta'(\varepsilon, y) = \delta(\varepsilon, y)$ by the definition of δ'. So $\delta(\varepsilon, y) = w_{m-2}$, and hence y is accepted by $\mathcal{A}$. It follows that $y \in \Sigma^* w$. This implies $x = yz \in \Sigma^* w \Sigma^*$.

Conversely, suppose $x \in \Sigma^* w \Sigma^*$. Write $x = ywz$ with y minimal. Since $yw \in \Sigma^* w$, we have $\delta(\varepsilon, yw) = w_{m-2}$. Furthermore, yw is the shortest prefix of x such that $\delta(\varepsilon, yw) = w_{m-2}$, since if there was a shorter prefix then y would not be minimal. This means that $\delta(\varepsilon, yw) = \delta'(\varepsilon, yw)$ by the definition of δ'. So $\delta'(\varepsilon, ywz) = w_{m-2}$ and hence $x = ywz$ is accepted by $\mathcal{A}'$. □

Fix w with state complexity m, and let $\mathcal{A}$ and $\mathcal{A}'$ be the DFAs for $\Sigma^* w$ and $\Sigma^* w \Sigma^*$, respectively, as described in the proof of Proposition 3. Fix T with state complexity at most n, and let $\mathcal{D}$ be an n-state DFA for T with state set Q_n and final state set F. The direct product DFA $\mathcal{A}' \times \mathcal{D}$ with final state set $\{w\} \times F$ recognizes $(\Sigma^* w \Sigma^*) \cap T$. Since $\mathcal{A}' \times \mathcal{D}$ has $(m-1)n$ states, this gives an upper bound of $(m-1)n$ on the state complexity of $(\Sigma^* w \Sigma^*) \cap T$.

Theorem 3. *Suppose $m \geqslant 3$ and $n \geqslant 2$. There exists a word w and a language T, with $\kappa(\{w\}) = m$ and $\kappa(T) = n$, such that $\kappa((\Sigma^* w \Sigma^*) \cap T) = (m-1)n$.*

Proof. Let $\Sigma = \{a, b\}$ and let $w = b^{m-2}$. Let $\mathcal{A}'$ be the DFA for $\Sigma^* w \Sigma^*$. Let T be the language of Definition 1. The DFA $\mathcal{A}' \times \mathcal{D}$ is illustrated in Figure 6.

We show that $\mathcal{A}' \times \mathcal{D}$ has $(m-1)n$ reachable and pairwise distinguishable states. For reachability, for $0 \leqslant i \leqslant m-2$ and $0 \leqslant q \leqslant n-1$, we can reach (b^i, q)

from the initial state $(\varepsilon, 0)$ by the word $a^q b^i$. For distinguishability, suppose we have states (b^i, q) and (b^j, q) in the same column q, with $i < j$. By b^{m-2-j} we reach $(b^{m-2+i-j}, q)$ and (w, q), with $b^{m-2+i-j} \neq w$. Then by a we reach (ε, qa) and (w, qa), which are distinguishable by a word in a^*. For states in different columns, suppose we have (b^i, p) and (b^j, q) with $p < q$. By a sufficiently long word in b^*, we reach (w, p) and (w, q). These states are distinguishable by a^{n-1-q}. So all reachable states are pairwise distinguishable. □

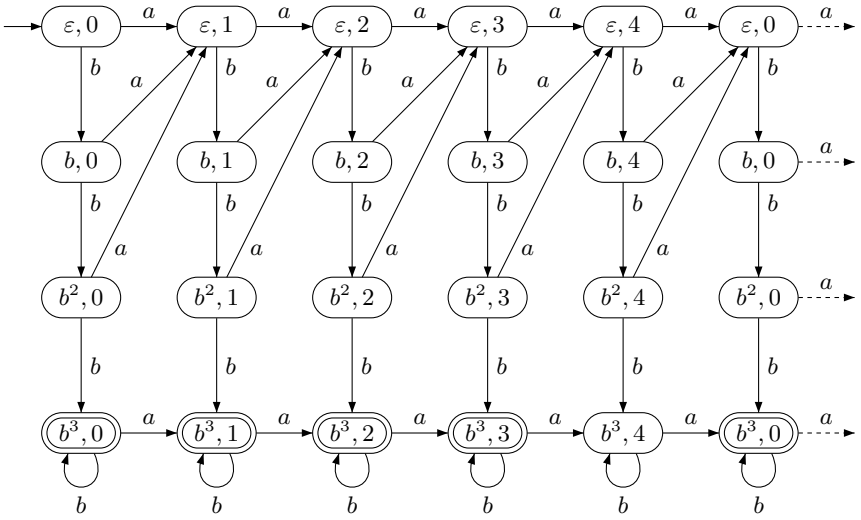

Fig. 6. DFA $\mathcal{A}' \times \mathcal{D}$ for matching a single factor, with $m = 5$ and $n = 5$. Column 0 is duplicated for a cleaner diagram; the DFA contains only one copy of this column.

6 Matching a Single Subsequence

Proposition 4. *If the state complexity of $\{w\}$ is m, then the state complexity of $\Sigma^* \sqcup w$ is $m - 1$.*

Proof. Define a DFA $\mathcal{A} = (W, \Sigma, \delta_{\mathcal{A}}, \varepsilon, \{w\})$ where $\delta_{\mathcal{A}}(w_i, a_{i+1}) = w_{i+1}$, and $\delta_{\mathcal{A}}(w_i, a) = w_i$ for $a \neq a_{i+1}$. Note that $\mathcal{A}$ is minimal: state w_i is reached by word w_i and states w_i, w_j with $i < j$ are distinguished by $a_{j+1} \cdots a_{m-2}$. We claim that $\mathcal{A}$ recognizes $\Sigma^* \sqcup w$.

Write δ rather than $\delta_{\mathcal{A}}$ to simplify the notation. Suppose $x \in \Sigma^* \sqcup w$. Then we can write $x = x_0 a_1 x_1 a_2 x_2 \cdots a_{m-2} x_{m-2}$, where $x_0, \ldots, x_{m-2} \in \Sigma^*$. We claim that $\delta(\varepsilon, x_0 a_1 x_1 \cdots a_i x_i) = w_j$ for some $j \geqslant i$. We proceed by induction on i. The base case $i = 0$ is trivial.

Now, suppose that $i > 0$ and $\delta(\varepsilon, x_0 a_1 x_1 \cdots a_{i-1} x_{i-1}) = w_j$ for some $j \geqslant i-1$. Then $\delta(\varepsilon, x_0 a_1 x_1 \cdots a_i x_i) = \delta(w_j, a_i x_i)$. We consider two cases:

- If $j = i - 1$, we have $\delta(w_{i-1}, a_i x_i) = \delta(w_i, x_i) = w_k$ for some k with $k \geqslant i$, as required.
- If $j > i - 1$, we have $\delta(w_j, a_i x_i) = w_k$ for some k with $k \geqslant i$, as required.

This completes the inductive proof. It follows then that $\delta(\varepsilon, x) = w_{m-2} = w$, and so x is accepted by $\mathcal{A}$. Conversely, if x is accepted by $\mathcal{A}$, then it is clear from the definition of the transition function that the letters $a_1, a_2, \ldots, a_{m-2}$ must occur within x in order, and so $x \in \Sigma^* \sqcup w$. □

Fix w with state complexity m, and let $\mathcal{A}$ be the DFA for $\Sigma^* \sqcup w$ described in the proof of Proposition 4. Fix T with state complexity at most n, and let $\mathcal{D}$ be an n-state DFA for T with state set Q_n and final state set F. The direct product DFA $\mathcal{A} \times \mathcal{D}$ with final state set $\{w\} \times F$ recognizes $(\Sigma^* \sqcup w) \cap T$. Since $\mathcal{A} \times \mathcal{D}$ has $(m-1)n$ states, this gives an upper bound of $(m-1)n$ on the state complexity of $(\Sigma^* \sqcup w) \cap T$.

Theorem 4. *Suppose $m \geqslant 3$ and $n \geqslant 2$. There exists a word w and a language T, with $\kappa(\{w\}) = m$ and $\kappa(T) = n$, such that $\kappa((\Sigma^* \sqcup w) \cap T) = (m-1)n$.*

Proof. Let $\Sigma = \{a, b\}$ and let $w = b^{m-2}$. Let $\mathcal{A}$ be the DFA for $\Sigma^* \sqcup w$. Let T be the language of Definition 1. The DFA $\mathcal{A} \times \mathcal{D}$ is illustrated in Figure 7.

We show that $\mathcal{A} \times \mathcal{D}$ has $(m-1)n$ reachable and pairwise distinguishable states. For reachability, for $0 \leqslant i \leqslant m-2$ and $0 \leqslant q \leqslant n-1$, we can reach (b^i, q) from the initial state $(\varepsilon, 0)$ by the word $a^q b^i$. For distinguishability, suppose we have states (b^i, q) and (b^j, q) in the same column q, with $i < j$. By b^{m-2-j} we reach $(b^{m-2+i-j}, q)$ and (w, q), with $b^{m-2+i-j} \neq w$. These states are distinguishable by a word in a^*. For states in different columns, suppose we have (b^i, p) and (b^j, q) with $p < q$. By a sufficiently long word in b^*, we reach (w, p) and (w, q). These states are distinguishable by a^{n-1-q}. So all reachable states are pairwise distinguishable. □

7 Conclusions

Building on previous work, we investigated the state complexity of "pattern matching" operations on regular languages, based on finding all words in a text language T which contain the single word w as either a prefix, suffix, factor, or subsequence. In all cases, the bounds were significantly lower than the general case, where w is replaced by a regular language P. Prefix matching is now linear in the input languages' state complexities, and the remaining cases are polynomial in the input state complexities. The general bounds were polynomial for prefix matching and exponential in the other cases. It is also worth noting that a binary alphabet is sufficient to reach all these bounds, including subsequence matching, whose bound was defined in terms of a growing alphabet in the general case. For languages with a unary alphabet, the state complexity was linear in all four cases.

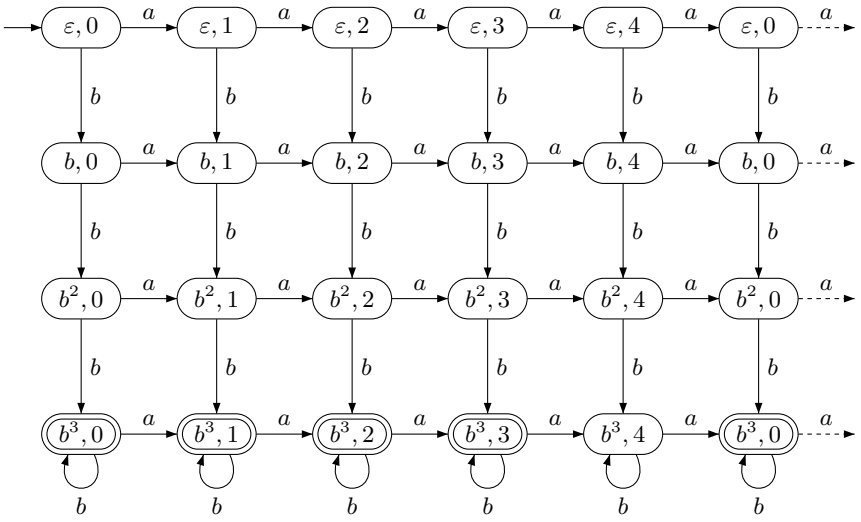

Fig. 7. DFA $\mathcal{A} \times \mathcal{D}$ for matching a single subsequence, with $m = 5$ and $n = 5$. Column 0 is duplicated for a cleaner diagram; the DFA contains only one copy of this column.

References

1. Brzozowski, J.A., Davies, S., Madan, A.: State complexity of pattern matching in regular languages. Theoret. Comput. Sci. (2018), https://doi.org/10.1016/j.tcs.2018.12.014
2. Brzozowski, J.A., Davies, S., Madan, A.: State complexity of pattern matching in regular languages (2018), `http://arxiv.org/abs/1806.04645`
3. Brzozowski, J.A., Jirásková, G., Li, B.: Quotient complexity of ideal languages. Theoret. Comput. Sci. 470, 36–52 (2013)
4. Crochemore, M., Hancart, C.: Automata for matching patterns. In: Rozenberg, G., Salomaa, A. (eds.) Handbook of Formal Languages, vol. 2, pp. 399–462. Springer (1997)
5. Elloumi, M., Iliopoulos, C., Wang, J.T., Zomaya, A.Y.: Pattern Recognition in Computational Molecular Biology: Techniques and Approaches. Wiley (2015)
6. Fu, Q., Lou, J.G., Wang, Y., Li, J.: Execution anomaly detection in distributed systems through unstructured log analysis. In: International Conference on Data Mining. pp. 149–158. IEEE (12 2009)
7. Gao, Y., Moreira, N., Reis, R., Yu, S.: A survey on operational state complexity. J. Autom. Lang. Comb. 21(4), 251–310 (2016)
8. Maslov, A.N.: Estimates of the number of states of finite automata. Dokl. Akad. Nauk SSSR 194, 1266–1268 (Russian). (1970), English translation: Soviet Math. Dokl. **11** (1970) 1373–1375
9. Yu, S., Zhuang, Q., Salomaa, K.: The state complexities of some basic operations on regular languages. Theoret. Comput. Sci. 125, 315–328 (1994)