State Complexity of Prefix Distance of Subregular Languages

Timothy Ng, David Rappaport, and Kai Salomaa(✉)

School of Computing, Queen's University, Kingston, ON K7L 3N6, Canada
{ng,daver,ksalomaa}@cs.queensu.ca

Abstract. The neighbourhood of a regular language of constant radius with respect to the prefix distance is always regular. We give upper bounds and matching lower bounds for the size of the minimal deterministic finite automaton (DFA) needed for the radius k prefix distance neighbourhood of an n state DFA that recognizes, respectively, a finite, a prefix-closed and a prefix-free language. For prefix-closed languages the lower bound automata are defined over a binary alphabet. For finite and prefix-free regular languages the lower bound constructions use an alphabet that depends on the size of the DFA and it is shown that the size of the alphabet is optimal.

1 Introduction

The neighbourhood of radius r of a language L consists of all strings that are within distance at most r from some string of L. A distance measure d is said to be regularity preserving if the neighbourhood of any regular language with respect to d is regular. Calude et al. [2] have shown that *additive distances* are regularity preserving. Additivity requires, roughly speaking, that the distance is compatible with concatenation of words in a certain sense and best known examples of additive distances include the Levenshtein distance and the Hamming distance [2,5].

The prefix distance of two words u and v is the sum of the lengths of the suffixes of u and v that begin after the longest common prefix of u and v. The suffix distance and the factor distance are defined analogously in terms of the longest common suffix (respectively, factor) of two words. It is known that the prefix, suffix and factor distance preserve regularity [4].

By the state complexity of a regularity preserving distance we mean the worst-case size of the minimal deterministic finite automaton (DFA) needed to recognize radius r neighbourhood of an n state DFA language (as a function of n and r). Tight bounds for the state complexity of prefix distance were recently obtained by the authors [14].

Worst-case state complexity bounds for general regular languages typically cannot be matched by finite languages, as first observed by Câmpeanu et al. [3], and the same holds for other proper sub-families of the regular languages.

C. Câmpeanu et al. (Eds.): DCFS 2016, LNCS 9777, pp. 192–204, 2016.
DOI: 10.1007/978-3-319-41114-9_15

Relations between different sub-regular language families have been investigated recently by Holzer and Truthe [11]. Bordihn et al. [1] have studied the state complexity of determinization of automata for the different sub-regular language families and further recent work on the state complexity of sub-regular language families has been done by Holzer et al. [8,10].

Here we study the state complexity of prefix distance for finite languages. Additionally, we concentrate on the classes of prefix-closed and prefix-free regular languages because their corresponding restricting properties can be viewed to be related to the definition of the prefix distance measure. We give tight state complexity bounds for the prefix distance of finite, prefix-closed and prefix-free regular languages. In the case of finite languages and prefix-free languages the lower bound construction uses an alphabet that depends linearly on the size of the DFA. We establish that the general upper bound cannot be matched by languages defined over an alphabet of smaller size.

2 Preliminaries

We briefly recall some definitions and notation used in the paper. For all unexplained notions on finite automata and regular languages the reader may consult the textbook by Shallit [15] or the survey by Yu [16]. A survey of distances is given by Deza and Deza [5]. Recent surveys on descriptional complexity of regular languages include [6,9,13].

In the following Σ is always a finite alphabet, the set of strings over Σ is Σ^* and ε is the empty string. The reversal of a string $x \in \Sigma^*$ is x^R. The set of nonnegative integers is $\mathbb{N}_0$. The cardinality of a finite set S is denoted $|S|$ and the powerset of S is 2^S. A string $w \in \Sigma^*$ is a *substring* or *factor* of x if there exist strings $u, v \in \Sigma^*$ such that $x = uwv$. If $u = \varepsilon$, then w is a *prefix* of x. If $v = \varepsilon$, then w is a *suffix* of x.

A *nondeterministic finite automaton* (NFA) is a 5-tuple $A = (Q, \Sigma, \delta, Q_0, F)$ where Q is a finite set of states, Σ is an alphabet, δ is a multi-valued transition function $\delta : Q \times \Sigma \to 2^Q$, $Q_0 \subseteq Q$ is a set of initial states, and $F \subseteq Q$ is a set of final states. We extend the transition function δ to a function $Q \times \Sigma^* \to 2^Q$ in the usual way. A string $w \in \Sigma^*$ is *accepted* by A if, for some $q_0 \in Q_0$, $\delta(q_0, w) \cap F \neq \emptyset$ and the language recognized by A consists of all strings accepted by A. An ε-NFA is an extension of an NFA where transitions can be labeled by the empty string ε [15,16], i.e., δ is a function $Q \times (\Sigma \cup \{\varepsilon\}) \to 2^Q$. It is known that every ε-NFA A has an equivalent NFA without ε-transitions and with the same number of states as A. An NFA $A = (Q, \Sigma, \delta, Q_0, F)$ is a *deterministic finite automaton* (DFA) if $|Q_0| = 1$ and, for all $q \in Q$ and $a \in \Sigma$, $\delta(q, a)$ either consists of one state or is undefined. Two states p and q of a DFA A are equivalent if $\delta(p, w) \in F$ if and only if $\delta(q, w) \in F$ for every string $w \in \Sigma^*$. A DFA A is *minimal* if each state $q \in Q$ is reachable from the initial state, a final state is reachable from each state q, and no two states are equivalent.

Note that our definition of a DFA allows some transitions to be undefined, that is, by a DFA we mean an incomplete DFA. It is well known that, for a regular

language L, the sizes of the minimal incomplete and complete DFAs differ by at most one. The constructions used in this paper are more convenient to formulate using incomplete DFAs but our results would not change in any significant way if we were to require that all DFAs are complete. The (incomplete deterministic) *state complexity* of a regular language L, $\mathrm{sc}(L)$, is the size of the minimal DFA recognizing L.

We define $\mathrm{pref}(L)$ to be the language of all prefixes of words belonging to L,

$$\mathrm{pref}(L) = \{u \in \Sigma^* \mid (\exists v \in \Sigma^*)\ uv \in L\}.$$

A language L is *prefix-closed* if $L = \mathrm{pref}(L)$. A language L is *prefix-free* if no word $u \in L$ is a proper prefix of any other word in L. A DFA A is *non-exiting* if a final state of A has no outgoing transitions. The minimal DFAs recognizing a prefix-free language have always the following property.

Lemma 1 ([7]). *If A is minimal and $L(A)$ is prefix-free, then A is non-exiting.*

To conclude this section, we recall definitions of the distance measures used in the following. Generally, a function $d : \Sigma^* \times \Sigma^* \to [0, \infty)$ is a *distance* if it satisfies for all $x, y, z \in \Sigma^*$, the conditions $d(x, y) = 0$ if and only if $x = y$, $d(x, y) = d(y, x)$, and $d(x, z) \leq d(x, y) + d(y, z)$. The *neighbourhood* of a language L of radius k with respect to a distance d is the set

$$E(L, d, k) = \{w \in \Sigma^* \mid (\exists x \in L)\ d(w, x) \leq k\}.$$

Let $x, y \in \Sigma^*$. The *prefix distance* of x and y counts the number of symbols which do not belong to the longest common prefix of x and y [4]. Formally, it is defined by

$$d_p(x, y) = |x| + |y| - 2 \cdot \max_{z \in \Sigma^*} \{|z| \mid x, y \in z\Sigma^*\}.$$

The state complexity of prefix distance was established in [14].

Theorem 1 ([14]). *For $n > k \geq 0$, if $\mathrm{sc}(L) = n$ then*

$$\mathrm{sc}(E(L, d_p, k)) \leq n \cdot (k + 1) - \frac{k(k+1)}{2}$$

and this bound can be reached in the worst case.

To conclude this section we recall from [14] the construction of a DFA that recognizes the prefix-distance neighbourhood of a regular language.

Let $A = (Q, \Sigma, \delta, q_0, F)$ be a DFA and $\varphi_A : Q \to \mathbb{N}_0$ be a function defined by

$$\varphi_A(q) = \min_{w \in \Sigma^*} \{|w| \mid \delta(q, w) \in F\}$$

The function $\varphi_A(q)$ gives the length of the shortest path from a state q to the closest reachable final state. Note that if $q \in F$, then $\varphi_A(q) = 0$.

We construct a DFA $A' = (Q', \Sigma, \delta', q_0', F')$ for the neighbourhood $E(L(A), d_p, k)$, $k \in \mathbb{N}$, as follows. We define the state set

$$Q' = ((Q - F) \times \{1, \ldots, k + 1\}) \cup F \cup \{p_1, \ldots, p_k\}. \tag{1}$$

The initial state q_0' is defined by

$$q_0' = \begin{cases} q_0, & \text{if } q_0 \in F; \\ (q_0, \varphi_A(q_0)) & \text{if } q_0 \notin F \text{ and } \varphi_A(q_0) \leq k; \\ (q_0, k+1) & \text{if } q_0 \notin F \text{ and } \varphi_A(q_0) > k. \end{cases}$$

The set of final states is given by

$$F' = ((Q - F) \times \{1, \ldots, k\}) \cup F \cup \{p_1, \ldots, p_k\}.$$

Let $q_{i,a} = \delta(i, a)$ for $i \in Q$ and $a \in \Sigma$, if $\delta(i, a)$ is defined. Then for all $a \in \Sigma$, the transition function δ' is defined for states $i \in F$ by

$$\delta'(i, a) = \begin{cases} (q_{i,a}, 1), & \text{if } q_{i,a} \in Q - F; \\ q_{i,a}, & \text{if } q_{i,a} \in F; \\ p_1, & \text{if } \delta(i, a) \text{ is undefined.} \end{cases}$$

For states $(i, j) \in Q - F \times \{1, \ldots, k+1\}$, δ' is defined

$$\delta'((i, j), a) = \begin{cases} q_{i,a}, & \text{if } q_{i,a} \in F; \\ (q_{i,a}, \min\{j+1, \varphi_A(q_{i,a})\}), & \text{if } \varphi_A(q_{i,a}) \text{ or } j+1 \leq k; \\ (q_{i,a}, k+1), & \text{if } \varphi_A(q_{i,a}) \text{ and } j+1 > k; \\ p_{j+1}, & \text{if } \delta(i, a) \text{ is undefined.} \end{cases}$$

Finally, we define δ' for states p_ℓ for $\ell = 1, \ldots, k-1$ by $\delta'(p_\ell, a) = p_{\ell+1}$.

The following Proposition 1 follows from the proof of Proposition 2 of [14]. Note that Proposition 2 of [14] establishes a stronger claim and the statement of the below proposition includes only the parts that we need in the later sections.

Proposition 1 ([14]). *(a) The DFA A' recognizes the neighbourhood $E(L(A), d_p, k)$.*

(b) The elements of the set $S_{ur} = \{(q, j) \mid q \in Q - F, 1 \leq j \leq k+1, j > \varphi_A(q)\}$ are all unreachable as states of the DFA A'.

3 Neighbourhoods of Finite Languages

We first consider the state complexity of neighbourhoods of finite languages with respect to the prefix distance.

Proposition 2. *Let L be a finite language recognized by a minimal DFA $A = (Q, \Sigma, \delta, q_0, F)$ with n states. Then*

$$\mathrm{sc}(E(L, d_p, k)) \leq (n-2) \cdot (k+1) - k^2 + 2.$$

Proof. We know that the neighbourhood of L of radius k with respect to the prefix distance is recognized by a DFA $A' = (Q', \Sigma, \delta', q_0'.F')$ obtained from A as in Proposition 1 where, furthermore, all elements of the set S_{ur} are unreachable. We show that there are more unreachable states in the case of finite languages.

Since A is acyclic, the number and length of words that reach each state $q \in Q$ is bounded. For $q \in Q$, let w_q denote the longest word that reaches q from the initial state q_0 without passing through a final state. Then for all states q with $|w_q| \leq k$, the states $(q, j) \in Q'$ with $j > |w_q|$ are unreachable as states of A' (where the set of states of A' is as in (1). That is, all states in the set

$$R_{ur} = \{(q, j) \mid q \in Q - F, 1 \leq j \leq k + 1, j > |w_q|\}$$

are unreachable in A'. By Proposition 1 (b) all elements of the set $S_{ur} = \{(q, j) \mid q \in Q - F, 1 \leq j \leq k + 1, j > \varphi_A(q)\}$ are also unreachable in A'. We note that increasing the number of final states of A by one decreases the cardinality of Q' by k and decreases the cardinality of S_{ur} and R_{ur} by at most k. However, we observe that A must have at least two final states to reach the bound. The last state of A, with no outgoing transitions, must be a final state since, otherwise, there are useless states. But this cannot be the only final state, since otherwise, for every state $q \in Q$ with $\varphi_A(q) > k$, only $(q, k + 1)$ is reachable. Thus, the initial state q_0 must also be a final state.

As in [14], we note that the cardinality of S_{ur} is minimized when exactly one non-final state has a shortest path of length i that reaches q_f. From the above it then follows that reaching the upper bound requires exactly two final states, one of which must be the initial state and the other which must have no outgoing transitions. Since A is acyclic, the initial state cannot have any incoming transitions, so the states in S_{ur} consist of those that can reach the non-initial final state, giving $\frac{k(k+1)}{2}$ unreachable states. Similarly, the cardinality of R_{ur} is minimized when exactly one non-final state has a longest word of length i which reaches it from q_0, giving $\frac{k(k+1)}{2}$ unreachable states.

Thus, the number of states of the minimal DFA for $E(L, d_p, k)$ is upper bounded by

$$(n-2)(k+1) + 2 + k - 2 \cdot \frac{k(k+1)}{2} = (n-2)(k+1) - k^2 + 2.$$

□

Next we give a lower bound construction that matches the upper bound of Proposition 2.

Lemma 2. *There exists a finite language recognized by a DFA with n states such that $E(L(A), d_p, k)$ requires at least $(n-2)(k+1) - k^2 + 2$ states.*

Proof. Let $A_n = (Q_n, \Sigma_n, \delta_n, q_0, F_n)$ where $Q_n = \{0, \ldots, n-1\}$, $\Sigma_n = \{a_1, \ldots, a_{n-3}\}$, $q_0 = 0$, $F_n = \{0, n-1\}$, and the transition function is defined by

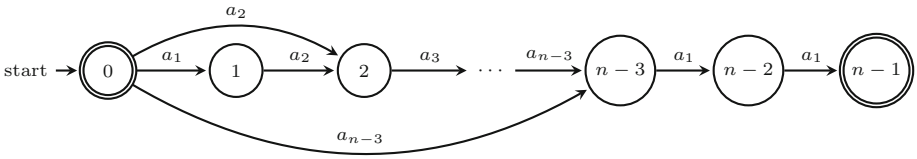

Fig. 1. The DFA A_n.

- $\delta_n(0, a_i) = i$ for $1 < j \leq n-3$,
- $\delta_n(i, a_{i+1}) = i+1$ for $0 \leq i < n-3$,
- $\delta_n(i, a_1) = i+1$ for $i = n-3, n-2$.

The DFA A_n is depicted in Fig. 1.

Let $A'_n = (Q'_n, \Sigma_n, \delta'_n, q'_0, F'_n)$ be the DFA constructed from A_n as in Proposition 1. First, we show that $(n-2)(k+1) - k^2 + 2$ states are reachable. States of the form p_i with $1 \leq i \leq k$ are reachable from states $0 \leq i \leq k$ on symbols a_j with $j \neq i+1$. For states of the form $(i,j) \in (Q_n - F_n) \times \{1, \ldots, k+1\}$, with $\varphi_{A_n}(i) > k$ and $j \leq i$, each (i,j) is reachable on the word $a_{i-j}a_{i-j+1}\cdots a_i$. However, states (i,j) with $j > \varphi_{A_n}(i)$ are unreachable by definition of A'_n and states (i,j) with $i < j \leq k$ are unreachable. Thus the number of unreachable states in $(Q_n - F_n) \times \{1, \ldots, k+1\}$ is

$$\sum_{i=n-k}^{n-1} |\{i\} \times \{\varphi_{A_n}(i)+1, \ldots, k+1\}| + \sum_{i=1}^{k} |\{i+1, \ldots, k+1\}|$$
$$= 2 \cdot \sum_{i=1}^{k} |\{i = 1, \ldots, k+1\}| = 2 \cdot \sum_{i=1}^{k} i = 2 \cdot \frac{k(k+1)}{2}.$$

Thus the number of reachable states is

$$(n-2)(k+1) - 2 + k - 2 \cdot \frac{k(k+1)}{2} = (n-2)(k+1) - k^2 + 2.$$

Now, we show that all reachable states are pairwise inequivalent.

- For states of the form p_i and p_j, $i < j$, the word a_1^{k-i} takes the machine from state p_i to p_k and is accepted. However, from state p_j, the word a_1^{k-i} reaches state p_k on the prefix a_1^{k-j} with no further transitions to read a_1^{j-i} and thus, the word is not accepted.
- For states of the form (i,j) and p_ℓ with $\ell < k$, we consider the word $z = w_i a_2^k$ with

  $$w_i = a_{n-i+1}a_{n-i+2}\cdots a_{n-3}a_1a_1.$$

  The prefix w_i takes the machine from state (i,j) to state $n-1$ and on the rest of the word a_2^k, the machine moves from $n-1$ to p_k and is accepted. However, from state p_ℓ, the computation on z reaches p_k before all of z is read, since $|z| = n - i + k > k - \ell$ and it is rejected.

– For states of the form (i, j) and (i', j') with $i < i'$ the states can be distinguished by $z = w_i a_2^k$ as above. For $i = i'$ and $j < j'$, let $z = a_i a_1^{k-j}$. From (i, j), the machine reads a_i and is taken to p_j, while from (i, j'), the machine is taken to $p_{j'}$. From above, p_j and $p_{j'}$ are distinguishable by a_1^{k-j}.

Thus, we have shown that there are $(n-2)(k+1) - k^2 + 2$ reachable states and that all reachable states are pairwise inequivalent.

□

Proposition 2 and Lemma 2 now yield a tight state complexity bound for the prefix distance neighbourhoods of regular languages.

Theorem 2. *Let L be a finite language. For $n > 2k \geq 0$, if $\mathrm{sc}(L) = n$, then*

$$\mathrm{sc}(E(L, d_p, k)) \leq (n-2) \cdot (k+1) - k^2 + 2,$$

and this bound can be reached in the worst case.

The lower bound construction of Lemma 2 uses, for a DFA with n states, an alphabet of cardinality $n - 3$. To conclude this section we show that the construction is optimal in the sense that the upper bound of Theorem 2 cannot be reached with an alphabet of cardinality less than $n - 3$.

Proposition 3. *Let A be a DFA recognizing a finite language with n states. If the state complexity of $E(L(A), d_p, k)$ equals $(n-2)(k+1) - k^2 + 2$, then the alphabet of A needs at least $n - 3$ letters.*

Proof. Let $A = (Q, \Sigma, \delta, q_0, F)$ with $|Q| = n$. Let $A' = (Q', \Sigma, \delta', q_0' F')$ be the DFA recognizing $E(L(A), d_p, k)$ constructed in Proposition 1. Recall from the proof of Proposition 2 that in order for A' to have the maximal number of states $(n-2)(k+1) - k^2 + 2$, a necessary condition is that $F = \{q_0, q_f\}$ and that there can be only one state q_1 with $\varphi_A(q_1) = 1$.

Now for all $q \in Q - \{q_0, q_f, q_1\}$, $\varphi_A(q) \geq 2$. By definition of the transition function δ', if $\varphi_A(q) \geq 2$, the state $(q, 1)$ can only be reached by a direct transition from a final state. Since q_f does not have any outgoing transitions, q_0 must have $n - 3$ outgoing transitions—one for each state q.

Furthermore, since A contains a final state q_f with no outgoing transitions, no additional symbols are required to reach p_1, as it can be reached from q_f via a direct transition on any symbol.

Since A is a DFA and q_0 has at least $n-3$ outgoing transitions, the cardinality of the alphabet must be at least $n - 3$. □

4 Neighbourhoods of Prefix-Closed and Prefix-Free Languages

Next, we consider the state complexity of neighbourhoods of prefix-closed and prefix-free regular languages with respect to the prefix distance.

Theorem 3. *Let L be a prefix-closed regular language recognized by an n-state DFA A. Then there is a DFA A' that recognizes the neighbourhood $E(L, d_p, k)$ with at most $n + k$ states and this bound is reachable.*

Proof. Since L is prefix-closed, every state of A must be an accepting state [12]. If A has n states, this means that the DFA A' constructed in Proposition 1 for the radius k neighbourhood has $n + k$ states.

We now define a prefix-closed regular language L_n such that a DFA recognizing $E(L_n, d_p, k)$ requires at least $n + k$ states. Let $L_n = \{a^i \mid 0 \leq i \leq n\}$. Then we define $A_n = (Q_n, \{a, b\}, \delta_n, q_0, F_n)$ where $Q_n = F_n = \{0, \ldots, n-1\}$, $q_0 = 0$, and the transition function δ_n is defined by $\delta_n(i, a) = i + 1$ for $0 \leq i \leq n - 1$.

Then we define the DFA recognizing $E(L_n, d_p, k)$ by $A' = (Q'_n, \{a, b\}, \delta'_n, q_0, F'_n)$ where $Q'_n = F'_n = Q_n \cup \{p_1, \ldots, p_k\}$ and the transition function defined by

- $\delta'_n(i, a) = i + 1$ for $0 \leq i < n - 1$,
- $\delta'_n(n - 1, a) = p_1$,
- $\delta'_n(i, b) = p_1$ for $0 \leq i < n - 1$,
- $\delta'_n(p_i, a) = \delta'_n(p_i, b) = p_{i+1}$ for $1 \leq i < k$.

Every state i, $0 \leq i \leq n - 1$, is reachable on the word a^i and every state p_i, $1 \leq i \leq k$ is reachable on the word b^i. The states $0 \leq i, i' \leq n-1$ are distinguished by the word b^{k-i} and the states p_i, p'_i, $1 \leq i, i' \leq k$ are also distinguished by the word b^{k-i}. The states i, $0 \leq i \leq n-1$ and p_j, $1 \leq j \leq k$ are distinguished by the word $a^{n-j}b^k$. Thus, there are $n + k$ reachable states and they are all pairwise distinguishable. □

Proposition 4. *Let L be a prefix-free regular language recognized by a minimal n-state DFA $A = (Q, \Sigma, \delta, q_0, F)$. Then there is a DFA B with at most $(n - 1)$ $k + 2 - \frac{k(k-1)}{2}$ states that recognizes the neighbourhood $E(L, d_p, k)$.*

Proof. Let $A' = (Q', \Sigma, \delta, q'_0, F')$ be the DFA constructed for the neighbourhood $E(L, d_p, k)$ as in Proposition 1. Since L is prefix-free, A must be non-exiting. That is, A has a single final state with no outgoing transitions. This property creates additional unreachable states in the DFA A' for $E(L, d_p, k)$.

For all non-final states $q \in Q - F$, the state $(q, 1)$ is reachable only if either $\varphi_A(q) = 1$ or there is a transition from a final state to q. However, since A is non-exiting, no final states may have any outgoing transitions, so the only states q where $(q, 1)$ is reachable are those with $\varphi_A(q) = 1$. However, for all such states q, the states (q, i) with $2 \leq i \leq k + 1$ are unreachable. Thus, to reach the upper bound on the number of states, the number of states q with $\varphi_A(q) = 1$ must be minimized if $k \geq 2$. If $k = 1$, then for each state $q \in Q - F$, either $(q, 1)$ is reachable or $(q, k + 1)$ is reachable, so the number of states with $\varphi_A(q) = 1$ need not be minimized.

By Proposition 1 (b) elements of the set $S_{ur} = \{(q, j) \mid q \in Q - F, 2 \leq j \leq k + 1, j > \varphi_A(q)\}$ are unreachable as states of A' (even without assuming that $L(A)$ is prefix-free. Let q_f be the sole final state of A. The set S_{ur} is minimized when exactly one non-final state q_i in the DFA A for each $1 \leq i \leq k$ has a shortest path of length i that reaches q_f. In this case, we have $|S_{ur}| = \frac{k(k-1)}{2}$.

Thus, in order to maximize the number of reachable states of A', the DFA A has a single final state and a single state q_1 with $\varphi_A(q_1) = 1$ if $k \geq 2$, giving us at most $(n-2)k + k + 2 - \frac{k(k-1)}{2} = (n-1)k + 2 - \frac{k(k-1)}{2}$ states of A' which are reachable. □

Next we present a lower bound construction that matches the bound of Proposition 4.

Lemma 3. *There exists a DFA A with n states recognizing a prefix-free regular language such that a DFA recognizing the neighbourhood $E(L(A), d_p, k)$ requires at least $(n-1)k + 2 - \frac{k(k-1)}{2}$ states.*

Proof. We define a DFA $A_n = (Q_n, \Sigma_n, \delta_n, q_0, F)$, shown in Fig. 2, by choosing

$$Q_n = \{0, \ldots, n-1\}, \Sigma_n = \{a_1, \ldots, a_{n-3}, b\},$$

$q_0 = 0$, $F = \{n-1\}$, and the transition function δ_n is given by

- $\delta_n(0, a_i) = i$ for $i = 1, \ldots, n-3$,
- $\delta_n(i, a_i) = i$ for $i = 1, \ldots, n-3$,
- $\delta_n(i, a_{i+1}) = i+1$ for $i = 1, \ldots, n-4$,
- $\delta_n(n-3, b) = n-2$, $\delta_n(n-2, b) = 0$, $\delta_n(0, b) = n-1$.

We transform A_n into the DFA $A'_n = (Q'_n, \Sigma_n, \delta'_n, q'_0, F')$ via the construction from Proposition 1. To determine the reachable states of Q'_n, we first note that the state $(0, 1)$ is reachable as it is the initial state. Note that the initial state

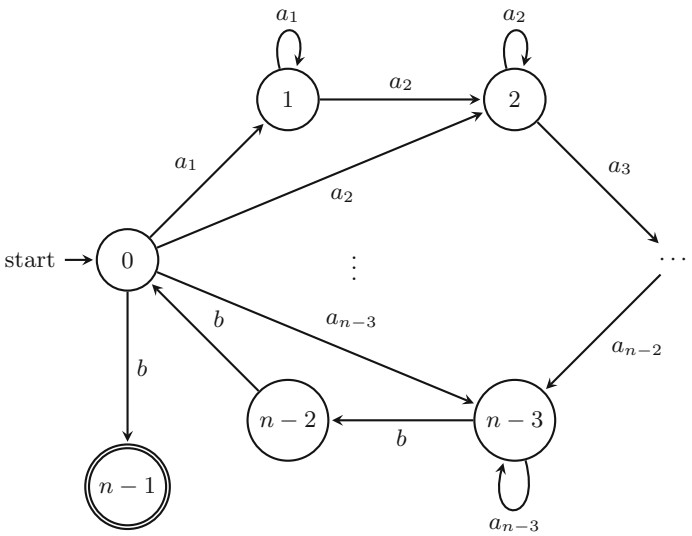

Fig. 2. The DFA A_n.

is $(0,1)$ since $\varphi_{A_n}(0) = 1$. The final state $n-1$ is reachable on the word b. Now consider states $p_1, \ldots, p_k$. The state p_ℓ is reachable on the word $b^{\ell+1}$ by first reading b to reach the final state and b^ℓ to reach the state p_ℓ.

Now consider states of the form $(i,j) \in (Q_n - \{0, n-1\}) \times \{2, \ldots, k+1\}$. Recall that states $(i,1)$ are unreachable for any state $i \in Q_n$ with $\varphi_{A_n} > 1$. Then for states $i \in Q_n$ with $\varphi_{A_n} > k$ and each $2 \leq j \leq k+1$, we can reach state (i,j) from $(0,1)$ via the word a_i^{j-1}. For states $i \in Q_n$ with $\varphi_{A_n} \leq k$, we can reach state (i,j) via the word a_i^{j-1} for $j = 2, \ldots, \varphi_{A_n}(i)$ and states (i,j) with $j > \varphi_{A_n}(i)$ are unreachable by definition of A'_n.

Finally, we can reach state $(n-2,2)$ via the word $a_{n-3}b$ and states $(n-2,j)$ are unreachable for $j > 2$ since $\varphi_{A_n}(n-2) = 2$. Thus the number of unreachable states in $(Q_n - \{0, n-1\}) \times \{2, \ldots, k+1\}$ is

$$\sum_{i=n-k}^{n-2} |\{i\} \times \{\varphi_{A_n}(i)+1, \ldots, k+1\}| = \sum_{i=1}^{k} |\{i+1, \ldots, k+1\}| = \sum_{i=1}^{k} i = \frac{k(k-1)}{2}.$$

Thus, the number of reachable states is

$$(n-2) \cdot k + 2 - \frac{k(k-1)}{2} + k = (n-1) \cdot k + 2 - \frac{k(k-1)}{2}.$$

Now, we show that all reachable states are pairwise inequivalent. First, note that as a final state of A, $n-1$ is not equivalent to a state of the form (i,j) in A'. Next, we distinguish states of the form (i,j) from states of the form p_ℓ. For each $1 \leq i \leq n-3$, reading the word a_i^k from state (i,j) takes the machine to state $(i, \min\{\varphi_A(i), k+1\})$. Then subsequently reading $a_{i+1}a_{i+2}\cdots a_{n-3}bbb$ takes the machine to the final state $n-1$. However, for every state p_ℓ, reading a_i^k forces the machine beyond state p_k, after which there are no transitions defined. The state $(n-2,2)$ is distinguished from all p_ℓ by the word b^{2+k}, $(0,1)$ by b^{1+k}, and $n-1$ by b^k.

Next, without loss of generality, let $\ell < \ell'$ and consider states p_ℓ and $p_{\ell'}$. Choose $z = b^{k-\ell}$. The string z takes state p_ℓ to the state p_k, where it is accepted. However, the computation on string z from state $p_{\ell'}$ is undefined since $\ell' + k - \ell > k$.

Finally, we consider states of the form (i,j). Let $i < i'$ and consider states (i,j) and (i',j'). Let $z = a_{i+1}a_{i+2}\cdots a_{n-3}bbbb^k$. From state (i,j), the word z goes to state $n-1$ on $a_{i+1}\cdots a_{n-3}bbb$. Then by reading b^k from state $n-1$, we reach state p_k, an accepting state. However, when reading z from state (i',j'), we immediately reach state $p_{j'+1}$ on a_{i+1}, since the transition on a_{i+1} is defined only for states $(0,1)$ and (i,j). Since the rest of the word z is of length greater than k, reading it takes us to state p_k with no further defined transitions for the rest of the word.

Next, consider the state (i,j) and (i,j'), where $j < j'$. First, consider the case when $\varphi_{A_n}(i) > k$. Then let $z = a_i^{k-j}$. Reading z from (i,j) takes us to state (i,k), which is a final state. However, from (i,j'), reading z brings us to state $(i,k+1)$ and so the computation is rejected.

Now, consider the case when $\varphi_{A_n}(i) \leq k$. Let $z = bb^{k-j-1}$. From state (i, j), reading b takes the machine to state p_{j+1} and reading b^{k-j-1} puts the machine in the accepting state p_k. However, reading z from (i, j') takes us to state p_k with $b^{j'-j}$ still unread since $j' + k - j - 1 > k$ and thus, with no further transitions available, the computation is rejected.

Thus, we have shown that there are $(n-1) \cdot k + 2 - \frac{k(k-1)}{2}$ reachable states and that all reachable states are pairwise inequivalent. □

Combining Proposition 4 and Lemma 3 we have:

Theorem 4. *Let L be a prefix-free regular language. For $n > k \geq 0$, if $\mathrm{sc}(L) = n$, then*

$$\mathrm{sc}(E(L, d_p, k)) \leq (n-1) \cdot k + 2 - \frac{k(k-1)}{2},$$

and this bound can be reached in the worst case.

The construction of Lemma 3 that establishes the lower bound for Theorem 4 uses an alphabet of size $n-2$, where n is the number of states of the DFA. The below result establishes that the size of the alphabet cannot be reduced.

Proposition 5. *Let A be a DFA recognizing a prefix-free regular language with n states. If the state complexity of $E(L(A), d_p, k)$ equals $(n-1)k + 2 - \frac{k(k-1)}{2}$, then the alphabet of A needs at least $n-2$ letters.*

Proof. Let $A = (Q, \Sigma, \delta, q_0, F)$ with $|Q| = n$. Let $A' = (Q', \Sigma, \delta', q_0' F')$ be the DFA recognizing $E(L(A), d_p, k)$ constructed in Proposition 1. Recall that as an automaton recognizing a prefix-free regular language A must be non-exiting. That is, A has a single final state q_f and it cannot have any outgoing transitions. Recall also from the proof of Proposition 4 that in order for A' to have the maximal number of states $(n-1)k + 2 - \frac{k(k-1)}{2}$, a necessary condition is that there can be only one state q_1 with $\varphi_A(q_1) = 1$ and one state q_2 with $\varphi_A(q_2) = 2$.

Now for all $q \in Q - \{q_f, q_1, q_2\}$, $\varphi_A(q) \geq 3$. Recall that since the sole final state q_f has no outgoing transitions, states $(q, 1)$ are reachable only if $\varphi_A(q) = 1$. Then by definition of the transition function δ', if $\varphi_A(q) \geq 3$, the state $(q, 2)$ can only be reached by a direct transition from a state q with $\varphi_A(q) = 1$. Thus, q_1 must have $n-2$ outgoing transitions—one for each state q with $\varphi_A(q) \geq 3$ and one additional transition to the final state q_f. Note that q_2 requires no direct transition from q_1 since $\varphi_A(q_2) = 2$ and thus $(q_2, 2)$ is the only reachable state of the form (q_2, j).

Furthermore, since A contains a final state q_f with no outgoing transitions, no additional symbols are required to reach p_1, as it can be reached from q_f via a direct transition on any symbol.

Since A is a DFA and q_1 has at least $n-2$ outgoing transitions, the cardinality of the alphabet must be at least $n-2$. □

5 Conclusion

We have given tight state complexity bounds for the prefix-distance neighbourhood of, respectively, finite, prefix-closed, and prefix-free languages. As can, perhaps, be expected the bound for prefix-closed languages is relatively easier to obtain and the matching lower bound construction uses a binary alphabet. The upper bound constructions for the finite and the prefix-free languages are more involved and the lower bound constructions use a variable size alphabet. Furthermore, we have shown that, in both cases, the alphabet size is optimal.

Since the reversal of a DFA is not, in general, deterministic, the state complexity bounds for suffix-distance (or factor-distance) neighbourhoods differ significantly from the corresponding bounds for prefix-distance neighbourhoods. Tight lower bounds are not known for suffix-distance neighbourhoods of general regular languages [14] or for various sub-regular language families. Such questions can be a topic for further research.

References

1. Bordihn, H., Holzer, M., Kutrib, M.: Determination of finite automata accepting subregular languages. Theor. Comput. Sci. **410**(35), 3209–3222 (2009)
2. Calude, C.S., Salomaa, K., Yu, S.: Additive distances and quasi-distances between words. J. Univ. Comput. Sci. **8**(2), 141–152 (2002)
3. Câmpeanu, C., Culik II, K., Salomaa, K., Yu, S.: State complexity of basic operations on finite languages. In: Boldt, O., Jürgensen, H. (eds.) WIA 1999. LNCS, vol. 2214, p. 60. Springer, Heidelberg (2001)
4. Choffrut, C., Pighizzini, G.: Distances between languages and reflexivity of relations. Theor. Comput. Sci. **286**(1), 117–138 (2002)
5. Deza, M.M., Deza, E.: Encyclopedia of Distances. Springer, Heidelberg (2009)
6. Gao, Y., Moreira, N., Reis, R., Yu, S.: A review on state complexity of individual operations. Faculdade de Ciencias, Universidade do Porto, Technical report DCC-2011-8. To appear in Computer Science Review. www.dcc.fc.up.pt/dcc/Pubs/TReports/TR11/dcc-2011-08.pdf
7. Han, Y.S., Salomaa, K., Wood, D.: State complexity of prefix-free regular languages. In: Proceedings of the 8th International Workshop on Descriptive Complexity of Formal Systems. pp. 165–176 (2006)
8. Holzer, M., Jakobi, S., Kutrib, M.: The magic number problem for subregular language families. Int. J. Found. Comput. Sci. **23**(1), 115–131 (2012)
9. Holzer, M., Kutrib, M.: Descriptional and computational complexity of finite automata – a survey. Inform. Comput. **209**, 456–470 (2011)
10. Holzer, M., Kutrib, M., Meckel, K.: Nondeterministic state complexity of star-free languages. In: Bouchou-Markhoff, B., Caron, P., Champarnaud, J.-M., Maurel, D. (eds.) CIAA 2011. LNCS, vol. 6807, pp. 178–189. Springer, Heidelberg (2011)
11. Holzer, M., Truthe, B.: On relations between some subregular language families. Proc. NCMA **2015**, 109–124 (2015)
12. Kao, J.Y., Rampersad, N., Shallit, J.: On NFAs where all states are final, initial, or both. Theor. Comput. Sci. **410**(47–49), 5010–5021 (2009)
13. Kutrib, M., Pighizzini, G.: Recent trends in descriptional complexity of formal languages. Bull. EATCS **111**, 70–86 (2013)

14. Ng, T., Rappaport, D., Salomaa, K.: State complexity of prefix distance. In: Drewes, F. (ed.) CIAA 2015. LNCS, vol. 9223, pp. 238–249. Springer, Heidelberg (2015)
15. Shallit, J.: A Second Course in Formal Languages and Automata Theory. Cambridge University Press, Cambridge (2009)
16. Yu, S.: Regular languages. In: Rozenberg, G., Salomaa, A. (eds.) Handbook of Formal Languages, pp. 41–110. Springer, Heidelberg (1997)