KLEENE CLOSURE AND STATE COMPLEXITY*

$Matúš Palmovský^1$

Abstract. We prove that the automaton presented by Maslov [Soviet Math. Doklady **11** (1970) 1373–1375] meets the upper bound $3/4 \cdot 2^n$ on the state complexity of Kleene closure. Our main result shows that the upper bounds $2^{n-1} + 2^{n-1-k}$ on the state complexity of Kleene closure of a language accepted by an *n*-state DFA with k final states are tight for every k with $1 \le k \le n$ in the binary case. We also study Kleene Closure on prefix-free languages. In the case of prefix-free languages, the Kleene closure may attain just three possible complexities n - 2, n - 1, and n. We give some survey of our computations.

Mathematics Subject Classification. 68Q19, 68Q45.

1. INTRODUCTION

Kleene closure is a basic operation on formal languages which is defined as $L^* = \{w \mid w = v_1 v_2 \cdots v_k, k \ge 0, v_i \in L \text{ for each } i\}$. It is known that if L is recognized by an n-state deterministic finite automaton (DFA), then the language L^* is recognized by a DFA of at most $3/4 \cdot 2^n$ states [13,18]. The first worst-case example meeting this upper bound was presented already by Maslov in 1970 [13]. However, he did a small error and did not give any proof in his paper.

Later *et al.* [18] proved that the size of the minimal DFA for Kleene closure depends on the number of final states in a given DFA, and that the upper bound is $2^{n-1} + 2^{n-1-k}$, where k is the number of final states (excluding the initial state, if it is also final).

In this paper we give a proof of Maslov's result and we fix an error in his paper [13] by proving that Maslov's automaton meets the upper bound $3/4 \cdot 2^n$. Then we show that the upper bounds $2^{n-1} + 2^{n-1-k}$ are tight for every n and k with $1 \le k \le n-1$. This is the main result of our paper. The witness automata are defined over a binary alphabet. The size of the alphabet is optimal since the state complexity of Kleene closure over a unary alphabet is only $(n-1)^2 + 1$ [18].

In the second part of our paper we consider not only the worst case, but rather study all possible values that can be obtained as the number of states of the minimal DFA recognizing the Kleene closure of a regular language represented by a minimal *n*-state DFA. The problem is known as "the magic number problem" in the literature, and so called "magic numbers" are exactly the "holes" in the hierarchy that cannot be obtained in such a way.

 $Keywords\ and\ phrases.$ Regular languages, finite automata, Kleene closure, state complexity.

^{*} Research supported by grant VEGA 2/0084/15.

¹ Mathematical Institute, Slovak Academy of Sciences, Grešákova 6, 040 01 Košice, Slovakia. palmovsky@saske.sk

M. PALMOVSKÝ

The problem was first stated for NFA to DFA conversion by Iwama *et al.* in [9]. It is known that in the ternary case, no magic numbers exist, that is, each value from n to 2^n may be obtained as the size of the minimal DFA equivalent to a given minimal *n*-state NFA [11]. On the other hand, it is known that in the unary case, magic numbers exist [6], but we do not know which values are magic. The binary case is still open.

For Kleene closure, the possible resulting vales are in the range from 1 to $3/4 \cdot 2^n$, for an alphabet of at least two symbols, and in the range from 1 to $(n-1)^2 + 1$ [3] for a unary alphabet, and it is known that for a growing alphabet of size 2n, no magic numbers exist [12]. On the other hand, in the unary case, two segments of magic numbers of length n are described in [3].

Here we study the binary case. Using the lists of pairwise non-isomorphic automata of 2, 3, 4, and 5 states, we compute the frequencies of the resulting complexities for Kleene closure, and show that every value in the range from 1 to $3/4 \cdot 2^n$ occurs at least ones. We display our results in diagrams, and compute the average complexity.

In the case of n = 6, 7, 8, 9, we change the strategy, and consider binary automata, in which the first symbol is a circular shift of the states, and the second symbol is generated randomly. We consider an arbitrary number of final states. We show that each value from 1 to $3/4 \cdot 2^n$ is attainable, and we show that for every m with $1 \le m \le 3/4 \cdot 2^n$, there exists an n-state binary DFA A such that the state complexity of $L(A)^*$ is exactly m. Thus our computations show, that in the binary case, up to n = 9, no magic numbers exists. Moreover, for every n, the numbers 1, n, and $2^{n-1} + 2^{n-1-k}$ with $1 \le k \le n-1$ are attainable by the complexity of Kleene closure. The situation is completely different in the case of a unary alphabet, where two holes of length n exist for every n [3].

In the last part of the paper, we study the same problem for the class of prefix-free languages. We show that the state complexity of the Kleene closure of a prefix-free language with state complexity n may attain just three possible values n - 2, n - 1, and n. We also present some results of our computations in the binary case.

2. Preliminaries

Let Σ be a finite alphabet and Σ^* the set of all strings over Σ . The empty string is denoted by ε . The length of a string w is |w|. A *language* is any subset of Σ^* . We denote the size of a set A by |A|, and its power-set by 2^A .

A deterministic finite state automaton is a quintuple $A = (Q, \Sigma, \delta, s, F)$, where Q is a finite set of states; Σ is a finite set of input symbols; δ is the transition function that takes as arguments a state and an input symbol and returns a state; s is an element of Q called the initial state; F is the set of final states (or accepting states), $F \subseteq Q$. The domain of δ is naturally extended from $Q \times \Sigma$ to $Q \times \Sigma^*$ as follows: $\delta(q, uv) = \delta(\delta(q, u), v)$ and $\delta(q, \varepsilon) = q$. The language accepted or recognized by the DFA A is defined as the set $L(A) = \{w \in \Sigma^* \mid \delta(s, w) \in F\}$.

A state q in Q is reachable if there exists a string w in Σ^* such that $\delta(s, w) = q$. Two states p and q are equivalent if for every string w in Σ^* , $\delta(p, w) \in F$ if and only if $\delta(q, w) \in F$. Two states are distinguishable if they are not equivalent.

Two automata are *equivalent* if they recognize the same language. A DFA A is *minimal* if every equivalent DFA has at least as many states as A. It is known that every regular language has a unique, up to isomorphism, minimal DFA, and that a DFA is minimal if and only if all its states are reachable, and no two distinct states are equivalent.

The state complexity of a regular language L, denoted by sc(L), is the number of states in the minimal DFA accepting the language L.

The state complexity of an operation on regular languages is the maximal state complexity of the language resulting from the operation as a function of the state complexities of the arguments [1,4,5]. Formally if f is a k-ary operation on regular languages over an alphabet Σ preserving regularity, then the state complexity of the operation f is given by function sc_f from \mathbb{N}^k to \mathbb{N} defined as $\mathrm{sc}_f(n_1, n_2, \ldots, n_k) = \max\{\mathrm{sc}(f(L_1, L_2, \ldots, L_k)) \mid L_i \subseteq \Sigma^* \text{ and } \mathrm{sc}(L_i) = n_i \text{ for } i = 1, 2, \ldots, k\}.$

A nondeterministic finite automaton is a quintuple $A = (Q, \Sigma, \delta, s, F)$, where Q, Σ, s , and F are the same as for a DFA, and δ is the transition function that takes a state in Q and an input symbol in Σ as arguments and returns a subset of Q. The language accepted or recognized by the NFA A is defined as the set $L(A) = \{w \in \Sigma^* \mid \delta(s, w) \cap F \neq \emptyset\}$.

Every NFA $A = (Q, \Sigma, \delta, s, F)$ can be converted to an equivalent DFA $A' = (2^Q, \Sigma, \delta', \{s\}, F')$, where $F' = \{R \subseteq Q \mid R \cap F \neq \emptyset\}$, and $\delta'(R, a) = \bigcup_{r \in R} \delta(r, a)$ for each R in 2^Q and each a in Σ [15]. We call the DFA A' the subset automaton of the NFA A. The subset automaton may not be minimal since some of its states may be unreachable or equivalent. To prove that the states of a subset automaton are not equivalent, we will use the following observation.

Proposition 2.1. Let $A = (Q, \Sigma, \delta, s, F)$ be an NFA. For every state q of A, let there exists a string w_q such that w_q is accepted by N only from the state q, that is, $\delta(q, w_q) \cap F \neq \emptyset$ and $\delta(p, w_q) \cap F = \emptyset$ if $p \neq q$. Then the subset automaton of the NFA A does not have equivalent states.

Proof. Let S, T be subsets of states of A, where $S \neq T$. Without loss of generality, there exists a state q such that $q \in S$ and $q \notin T$. Then the string w_q is accepted from S but w_q is rejected from T. Hence S and T are not equivalent.

Definition 2.2. For languages K and L the concatenation $K \cdot L$ is defined as the language $K \cdot L = \{uv \mid u \in K, v \in L\}$. The language L^i with $i \ge 0$ is defined inductively by $L^0 = \{\varepsilon\}$, $L^1 = L$, $L^{i+1} = L^i \cdot L$. The Kleene closure of a language L is the language L^* defined as

$$L^* = \bigcup_{i \ge 0} L^i.$$

3. STATE COMPLEXITY OF KLEENE CLOSURE

First, we recall the standard construction of a nondeterministic automaton recognizing the Kleene closure of a given language represented by DFA.

Construction 1. Let $A = (Q, \Sigma, \delta, s, F)$ be a DFA accepting a language L. Construct an NFA A^* for the language L^* from the DFA A as follows:

- All original transitions of A are preserved in A^{*}.
- For each state q in Q and each symbol a in Σ such that $\delta(q, a) \in F$, add the transition on a from q to s.
- Add a new initial state q_0 to Q and make this state accepting. For each symbol a in Σ , add the transition on a

from q_0 to $\delta(s, a)$ if $\delta(s, a) \notin F$, and from q_0 to $\delta(s, a)$ and from q_0 to s if $\delta(s, a) \in F$.

• The final states of A^* are $F \cup \{q_0\}$.

The following upper bound on the state complexity of the Kleene closure of a language represented by an n-state DFA with k final states is from [18]. For the sake of completeness, we give a simplified proof here.

Lemma 3.1 (Upper bound [18]). Let $A = (Q, \Sigma, \delta, s, F)$ be an n-state DFA such that $|F \setminus \{s\}| = k$. Then the minimal DFA for the language $L(A)^*$ has at most $2^{n-1} + 2^{n-1-k}$ states.

Proof. Construct the NFA A^* for the language $L(A)^*$ as described above. Consider the deterministic subset automaton of the NFA A^* . Let S be a reachable state in this DFA. Notice that if a final state of the NFA A^* is in S, then the state s is also in S. The empty set is unreachable since the DFA A is deterministic and complete.



FIGURE 1. YZS'94 automaton meeting the bound $3/4 \cdot 2^n$ for Kleene closure.



FIGURE 2. The Maslov's DFA A.



FIGURE 3. An NFA A^* for $L(A)^*$.

It follows that only the following subsets can be reachable in the subset automaton:

- (1) $\{q_0\};$
- (2) $S \subseteq Q$ with $s \in S$;
- (3) $S \subseteq Q \setminus (F \cup \{s\})$ and $S \neq \emptyset$.

Hence we get at most $1 + 2^{n-1} + 2^{n-1-k} - 1$ reachable sets, which gives the desired upper bound.

Notice that the number $2^{n-1}+2^{n-1-k}$ is maximal if k = 1. For k = 1, we have $2^{n-1}+2^{n-1-k} = 2^{n-1}+2^{n-2} = 3/4 \cdot 2^n$. Thus we get the following upper bound.

Corollary 3.2. Let L be a language accepted by an n-state DFA. Then the minimal DFA for the language L^* has at most $3/4 \cdot 2^n$ states.

Yu et al. [18] presented for each $n \ge 2$ the witness language accepted by DFA shown in Figure 1, and they proved that it meets the upper bound $3/4 \cdot 2^n$.

Maslov [13] claimed that the state cost of the Kleene closure is $3/4 \cdot 2^n - 1$ and that the DFA in Figure 2 meets this bound. The claim was not supported by any proofs and, actually, it is wrong. In fact, from the already discussed results by Yu, Zhuang and Salomaa, the state cost is $3/4 \cdot 2^n$. Actually, even the DFA in Figure 2 meets this bound, as we now show.

First, construct an NFA A^* for the language $L(A)^*$ by adding the transition on a from n-2 to 0, by adding a new initial and final state q_0 , and by adding the transition on a from q_0 to 1 and the transition on b from q_0 to 0. The NFA A^* is shown in Figure 3. We are going to show that the subset automaton of the NFA A^* has at least $3/4 \cdot 2^n$ reachable and pairwise distinguishable states. To this aim let \mathcal{R} be the following family of subsets

 $\mathcal{R} = \{\{q_0\}\} \cup \{S \subseteq \{0, 1, \dots, n-1\} \mid S \notin \emptyset \text{ and if } n-1 \in S, \text{ then } 0 \in S\}.$

Lemma 3.3. Let $n \ge 3$. Each nonempty subset S in \mathcal{R} is reachable in the subset automaton of the NFA A^* shown in Figure 3.

Proof. By induction on |S|, we prove that every subset S in \mathcal{R} is reachable. The base is |S| = 1. The set $\{q_0\}$ is reachable since it is the initial state of the subset automaton. The set $\{i\}$, where $0 \le i \le n-2$, is reached from

 $\{q_0\}$ by the string ba^i since we have $\{q_0\} \xrightarrow{b} \{0\} \xrightarrow{a^i} \{i\}$. Assume that every set S in \mathcal{R} with |S| = k, where $1 \le k \le n-1$, is reachable. Let $S = \{i_1, i_2, i_3, \ldots, i_k, i_{k+1}\}$, where $0 \le i_1 < i_2 < \cdots < i_k < i_{k+1} \le n-1$, be a set in \mathcal{R} of size k+1. Consider three cases:

- (i) $i_1 = 0$ and $i_{k+1} = n 1$. Take $S' = \{i_2 1, i_3 1, \dots, i_k 1, n 2\}$. Then $S' \in \mathcal{R}$ since $n 1 \notin S'$. Next, |S'| = k and therefore S' is reachable by the induction hypothesis. Since $S' \xrightarrow{a} \{0, i_2, i_3, \dots, i_k, n 1\} = S$, the set S is reachable;
- (ii) $i_1 = 0$ and $i_{k+1} < n-1$. Take $S' = \{0, i_2 + x, i_3 + x, \dots, i_k + x, n-1\}$, where $x = n-1-i_{k+1}$. Then |S'| = k+1 and S' contains states 0 and n-1. Therefore, the set S' is reachable as shown in case (i). Since $S' \xrightarrow{b^x} \{0, i_2, i_3, \dots, i_k, i_{k+1}\} = S$, the set S is reachable;
- (iii) $i_1 > 0$. Then $i_{k+1} < n-1$ since the set S is in the family \mathcal{R} . Take $S' = \{0, i_2 i_1, i_3 i_1, \dots, i_k i_1, i_{k+1} i_1\}$. Then |S'| = k + 1 and S' contains the state 0. Therefore the set S' is reachable as shown in cases (i) and (ii). Since we have $|S'| \stackrel{a^{i_1}}{\longrightarrow} \{i_1, i_2, i_3, \dots, i_k, i_{k+1}\} = S$, the set S is reachable.

Our proof is complete.

Lemma 3.4. Let $n \ge 3$. All the subsets in \mathcal{R} are pairwise distinguishable in the subset automaton of the NFA A^* shown in Figure 3.

Proof. Notice that the string a^{n-1-i} is accepted by the NFA A^* only from the state *i*. Similarly as in the proof of Proposition 2.1, we can show that no two distinct subsets of $\{0, 1, \ldots, n-1\}$ are equivalent.

Next, we need to show that $\{q_0\}$ and each final subset S of $\{0, 1, \ldots, n-1\}$ are distinguishable. If S is a final subset, then $n-1 \in S$. Consider the string a^n . The set $\{q_0\}$ goes on a^n to $\{0, 1\}$, which is a non-final set since $n \geq 3$. However, in the NFA A^* , a computation on a^n starting in the state n-1 can reach the state n-1. It follows that a^n is accepted by the subset automaton from S. Thus the string a^n distinguishes $\{q_0\}$ and S. \Box

Hence we have shown that the minimal DFA for L(A*) has at least $3/4 \cdot 2^n$ states, which combined with the upper bound in Corollary 3.2, gives the following result.

Theorem 3.5. Let L be the language accepted by the Maslov's automaton shown in Figure 2. Then the minimal DFA for the language L^* has $3/4 \cdot 2^n$ states.

4. Kleene closure for automata with k final states

Notice that the upper bound given by Lemma 3.1 depends on the number of final states in a given DFA. Now, in the main result of our paper, we present binary automata with k final states that meet the upper bound $2^{n-1} + 2^{n-1-k}$, where $1 \le k \le n-1$.

To this aim, consider an *n*-state DFA $A = (Q, \Sigma, \delta, s, F)$, where

- $Q = \{0, 1, \dots, n-1\};$
- $\Sigma = \{a, b\};$



FIGURE 4. Modified YZS'94 automaton; k = 3.



FIGURE 5. The NFA N for modified YZS'94 automaton; k = 3.

- s = 0;
- $F = \{n k, n k + 1, n k + 2, \dots, n 1\};$ • $\delta(i, a) = (i + 1) \mod n,$ $\delta(0, b) = 0,$
- $$\begin{split} \delta(i,b) &= i+1 \text{ if } 1 \leq i \leq n-3, \\ \delta(n-2,b) &= 0, \\ \delta(n-1,b) &= n-1. \end{split}$$

The DFA A with k = 3 is shown in Figure 4. Notice that this automaton is obtained by a modification of YZS'94 automaton in Figure 1 presented in [18]. The transitions in these two automata differ only in transitions on b in the states n - 2 and n - 1.

Lemma 4.1. The DFA A described above is minimal.

Proof. Two distinct final states i and j, where i < j, can be distinguished by the string in a^{n-j} . Two distinct non-final states i and j, where i < j, can be distinguished by the string a^{n-k-j} .

Construct an NFA A^* for the language $L(A)^*$ as described in Construction 1. For k = 3, the NFA A^* is shown in Figure 5. Consider the subset automaton of A^* , and let us show that this subset automaton has $2^{n-1} + 2^{n-1-k}$ reachable and pairwise distinguishable states.

Lemma 4.2. The subset automaton of the NFA A^* , shown in Figure 5, has at least $2^{n-1} + 2^{n-1-k}$ reachable state.

Proof. Notice that if a reachable set contains a final state of A^* , then it must contain also the state 0. Let

 $\mathcal{R} = \{\{q_0\}\} \cup \{S \subseteq \{0, 1, \dots, n-1\} \mid S \notin \emptyset \text{ and if } S \cap F \neq \emptyset, \text{ then } 0 \in S\}$

be a family of $2^{n-1} + 2^{n-1-k}$ subsets of $\{q_0\} \cup \{0, 1, \ldots, n-1\}$. Let us show that each subset S in \mathcal{R} is reachable in the subset automaton of A^* . The set $\{q_0\}$ is reachable since it is the initial state of subset automaton. The set

 $\{0\}$ is reached from $\{q_0\}$ by b, and since we have $\{0\} \xrightarrow{a^i} \{i\}$ if $1 \le i \le n-k-1$, all subsets S in \mathcal{R} with |S| = 1 are reachable. Next we have

$$\{n-k-1\} \xrightarrow{a} \{0, n-k\} \xrightarrow{b^{i}} \{0, n-k+i\} \text{ for } i = 1, 2, \dots, k-2, \\ \{0, n-2\} \xrightarrow{a} \{0, 1, n-1\} \xrightarrow{b^{n-3}} \{0, n-2, n-1\} \xrightarrow{b} \{0, n-1\}, \\ \{0, n-1\} \xrightarrow{a} \{0, 1\} \xrightarrow{b^{i-1}} \{0, i\}$$

for $1, 2, \ldots, n - k - 1$.

Finally, if $1 \le i < j \le n - k - 1$, then $\{i, j\}$ is reached from $\{0, j - i\}$ by a^i . Thus each S in \mathcal{R} with |S| = 2 is reachable.

Assume that every set S in \mathcal{R} with |S| = t, where $2 \le t \le n-1$, is reachable. Let $S = \{i_1, i_2, \ldots, i_t, i_{t+1}\}$, where $0 \le i_1 < i_2 < \ldots < i_t < i_{t+1} \le n-1$, be a set in \mathcal{R} of size t + 1. Consider three cases:

(i) $i_1 = 0$ and $i_{t+1} = n-1$. Let $S' = \{0, i_3 - i_2, i_4 - i_2, \dots, i_t - i_2, n-2\}$. Then $S' \in \mathcal{R}$ since $0 \in S'$. Moreover, the set S' is reachable by the induction hypothesis, using |S'| = t. But then

$$S' \xrightarrow{a} \{0, 1, i_3 - i_2 + 1, \dots, i_t - i_2 + 1, n - 1\} \xrightarrow{b^{i_2 - 1}} S,$$

and hence the set S is also reachable.

- (ii) $i_1 = 0$ and $i_{t+1} < n-1$. Let $S' = \{0, i_3 i_2, \dots, i_{t+1} i_2, n-1\}$. Then S' is of size t+1 and contains 0 and n-1, thus S' is reachable by (i). Since $S' \xrightarrow{a} \{0, 1, i_3 i_2 + 1, \dots, i_t i_2 + 1, i_{t+1} i_2 + 1\} \xrightarrow{b^{i_2-1}} S$, the set S is reachable.
- (iii) $i_1 > 0$. Since the set S is in the family \mathcal{R} , we must have $i_{t+1} < n-k$. Take $S' = \{0, i_2 i_1, i_3 i_1, \dots, i_t i_1, i_{t+1} i_1\}$. Then |S'| = t + 1 and S' contains state 0. Therefore the set S' is reachable as shown in cases (i) and (ii). Since we have $S' \xrightarrow{a^{i_1}} \{i_1, i_2, i_3, \dots, i_t, i_{t+1}\} = S$, the set S is reachable.

This proves the reachability of $2^{n-1} + 2^{n-1-k}$ states.

Lemma 4.3. All reachable states of the subset automaton of the NFA A^* , shown in Figure 5, are pairwise distinguishable.

Proof. Notice that in the NFA A^* , the string b^n is accepted only from the state n - 1. Since there is a unique transition on a going to the state n - 1, which goes from the state n - 2, the string ab^n is accepted by A^* only from the state n - 2. Next, notice that each transition from i to i + 1 on a is a unique transition on a going to the state i + 1. It follows that the string $a^{n-1-i}b^n$ is accepted by A^* only from the state i ($0 \le i \le n - 2$). Similarly as in the proof of Proposition 2.1, this proves the distinguishability of subsets of $\{0, 1, \ldots, n - 1\}$. Since q_0 is a final state in A^* , the set $\{q_0\}$ is not equivalent to any S not containing a final state of A^* . Now we need to show that $\{q_0\}$ is not equivalent to any final subset S of $\{0, 1, \ldots, n - 1\}$. If S is final, then there is a final state $i \ge n - k$ such that $i \in S$. Then $a^{n-1-i}b^n$ is accepted by the subset automaton from S and rejected from $\{q_0\}$. This concludes the proof.

Now, we can state our main result.

Theorem 4.4. Let $n \ge 3$ and $1 \le k \le n-1$. There exists a binary n-state DFA A with k final states such that the minimal DFA for the language $L(A)^*$ has $2^{n-1} + 2^{n-1-k}$ states.

Proof. Let A be a DFA described on page 255. By Lemmata 4.2 and 4.3, the minimal DFA for $L(A^*)$ has at least $2^{n-1} + 2^{n-1-k}$ states, which combined with the upper bound in Lemma 3.1, proves the theorem.

M. PALMOVSKÝ

5. The range of complexities for Kleene closure

In this section, we consider not only the worst case, but rather study all possible values that can be obtained as the number of states of the minimal DFA recognizing the Kleene closure of a regular language represented by a minimal *n*-state DFA.

For Kleene closure, the possible resulting values are in the range from 1 to $3/4 \cdot 2^n$, and it is known that for a growing alphabet of size 2^n , no gaps in the hierarchy of possible complexities exist [10]. Recently, the alphabet of size 2n has been used to produce the whole range of complexities [12]. Here we study possible value for the binary case. We do not solve this problem completely, but we have some partial results. It follows from the previous section that each value $2^{n-1} + 2^{n-1-k}$ for k = 1, 2, ..., n-1 is attainable for a binary alphabet. The next two propositions show that the values 1 and n are attainable in the binary case.

Proposition 5.1. For every n, there exists a binary language L accepted by a minimal n-state DFA such that the language L^* has state complexity 1.

Proof. If n = 1, then take $L_1 = \{a, b\}^*$. If n = 2, take $L_2 = (a + b)(a + b)^*$. If n = 3, take $L_3 = a + b$. If $n \ge 4$, take $L_n = \{a, b\} \cup \{w \mid |w| \ge n - 1\}$. Then the minimal DFA for L_n has n states. Since $a \in L_n, b \in L_n$, we have $L_n^* = \{a, b\}^*$, and therefore the state complexity of L_n^* is 1.

Proposition 5.2. For every n, there exists a binary language L accepted by a minimal n-state DFA such that the language L^* has state complexity n.

Proof. Let $L = ((a+b)^n)^*$. The minimal DFA for L has n states. Next, we have $L = L^*$ and therefore the state complexity of L^* is n.

We did some computations concerning the Kleene closure on binary regular languages. We found out that there are no holes in the hierarchy up to n = 9, even in the binary case. Our experimental observations suggest that the average state complexity approximates n in limit. Our results can be found at http://im.saske.sk/~palmovsky/Kleene%20Closure

6. KLEENE CLOSURE ON PREFIX-FREE LANGUAGES

In this section we study the Kleene closure operation on prefix-free languages. First, let us recall some definitions.

The string u is a *prefix* of w if w = uv for some v. If, moreover, the string v is non-empty, then u is a proper prefix of w. A language is *prefix-free* if it does not contain two distinct strings, one of which is a prefix of the other. The following characterization of minimal DFAs accepting prefix-free languages is well known.

Proposition 6.1 ([8]). Let A be a minimal DFA for a non-empty language L. Then L is prefix-free if and only if A has exactly one final state with transitions going to the dead state on each symbol of the input alphabet.

It should be pointed out that this does not exclude the possibility that the dead state is also reached by transitions from some other (non-final) states. Using this characterization, a DFA B for the Kleene closure of a prefix-free language L, accepted by an n-state DFA $A = (Q, \Sigma, \delta, s, \{f\})$, can be constructed as follows. We make the final state f initial, and we redirect the transition on each symbol a from the final state f to the state $\delta(s, a)$ [8]. Formally, we have the following construction.

Construction 2. Let $A = (Q, \Sigma, \delta, s, \{f\})$ be a DFA accepting a prefix-free language L. Construct a DFA $A^* = (Q, \Sigma, \delta_B, f, \{f\})$ for L^* , where

$$\delta^*(q,a) = \begin{cases} \delta(q,a), & \text{if } q \neq f, \\ \delta(s,a), & \text{if } q = f. \end{cases}$$

This construction gives an *n*-state DFA A^* for the language L^* . The aim of this section is to show that the resulting complexity of L^* may be n - 2, n - 1 or n. Let us start with the following observation.

Lemma 6.2. Let A be a minimal DFA for a prefix-free language with the final state f and the dead state d. Let p and q be two distinct non-final states different from d. Then p and q can be distinguished by a string w such that the computations of A on w starting in the states p and q do not use any transition from f to d.

Proof. Let δ be the transition function of A. Let a string w be accepted from p and rejected from q. Then the computation on w from p cannot use any transition from f to d, otherwise w would be rejected from p. If the computation on w from q uses a transition from f to d on a symbol a, then w can be factorized as w = uav such that $\delta(q, u) = f$, $\delta(f, a) = d$, and $\delta(d, v) = d$. Hence u is accepted from q. Consider the computation on u from p. Since u is a proper prefix of w and w is accepted from p, this computation does not use any transition from f to d. Next, this computation must be rejecting because otherwise w would be rejected from p. Thus u is the desired string, and the proof is complete.

Now we are ready to prove the following result.

Lemma 6.3. Let L be a prefix-free regular language with sc(L) = n, where $n \ge 4$. Then $n - 2 \le sc(L^*) \le n$.

Proof. Let $A = (\{1, 2, ..., n-2, f, d\}, \Sigma, \delta, 1, \{f\})$ be the minimal DFA for a prefix-free language L with the dead state d. Construct a DFA A^* as described in Construction 2

Since L is prefix-free, each transition from the final state f goes to the dead state d.

In DFA A, the states 1 and d may be unreachable. Consider a state i with $i \neq 1$ and $i \neq d$, and let us show that it is reachable in A^* . Since A is minimal, the state i is reachable in A, that is, there is a non-empty string w such that $i = \delta(1, w)$. Let w = av. Then in A^* we have $f \xrightarrow{a} \delta(1, a) \xrightarrow{v} i$. Hence i is reachable in A^* .

Next, in DFA A^* , the state d is the only state that does not accept any string, and the state f is the only final state. If the state 1 is reachable in A^* , then the non-final states in $\{1, 2, \ldots, n-2\}$ are distinguishable by Lemma 6.2. Now, assume that the state 1 is unreachable in A^* . This means that the state 1 does not have any in-transition in A. It follows that no out-transition from the state 1 can be used to show the distinguishability of the states $2, 3, \ldots, n-2$ in DFA A. Therefore these states are distinguishable in DFA A^* by Lemma 6.2. Hence all states of A^* are pairwise distinguishable, and our proof is complete.

Theorem 6.4. Let $n \ge 4$ and $n - 2 \le \alpha \le n$. There exists a binary prefix-free language L such that sc(L) = n and $sc(L^*) = \alpha$.

Proof. Consider the minimal DFAs A, B, and C shown in Figure 6 top, middle, and bottom, respectively. All these automata accept prefix-free languages. Construct DFAs A^* , B^* , and C^* as described in Construction 2. Notice that each state in $\{1, 2, \ldots, n-2\} \cup \{f, d\}$ is reachable in DFA A^* , the state d is unreachable in B^* , and the states 1 and d are unreachable in C^* . By Lemma 6.2, all reachable states are distinguishable in DFAs A^* , B^* , and C^* . Hence the Kleene closures of languages L(A), L(B), and L(C) meet complexities n, n-1, and n-2, respectively.

Let us denote by $R_k(n)$ the set of possible complexities of the Kleene closure of prefix-free languages with state complexity n over a k-letter alphabet, that is,

$$R_k(n) = \{ \operatorname{sc}(L^*) \mid L \subseteq \Sigma^*, |\Sigma| = k, L \text{ is prefix-free and } \operatorname{sc}(L) = n \}$$

Using this notation, we get the following result.

Theorem 6.5. Let $R_k(n)$ be the set of possible complexities of the Kleene closure of prefix-free languages over a k-letter alphabet, as defined above. Then we have

(a) $R_k(1) = R_k(2) = \{2\}$ where $k \ge 1$;



FIGURE 6. The DFAs of the prefix-free languages meeting the complexities n (top), n-1 (middle), and n-2 (bottom) for their Kleene closure.

- (b) $R_1(n) = \{n-2\}$ where $n \ge 3$;
- (c) $R_2(3) = \{1, 2\}$ and $R_k(3) = \{1, 2, 3\}$ if $k \ge 3$;
- (d) $R_k(n) = \{n-2, n-1, n\}$ if $k \ge 2$ and $n \ge 4$.

Proof.

- (a) The only prefix-free languages with state complexity 1 and 2 are the empty language and the language $\{\varepsilon\}$, respectively. The Kleene closure of both these languages is $\{\varepsilon\}$, with $sc(\{\varepsilon\}) = 2$ for $|\Sigma| \ge 1$.
- (b) The only prefix-free unary language with the state complexity n, where $n \ge 3$, is the language $\{a^{n-2}\}$. Its Kleene closure is the language $(a^{n-2})^*$ with the state complexity n-2.
- (c) Using a binary alphabet and taking into account Proposition 6.1, we cannot reach both the initial and the dead states of a 3-state DFA A from the initial state. Therefore, in the DFA A^* , at most two states are reachable. The languages b^*a and a + b meet the complexities 2 and 1, respectively. The language b^*a over the ternary alphabet $\{a, b, c\}$ meets the complexity 3.

(d) The equality is given by Lemma 6.3 and Lemma 6.4.

We also did some computations. Having the lists of binary minimal and pairwise non-isomorphic prefixfree DFAs, we computed the complexities of the Kleene closure of the accepted languages. The frequencies of complexities n - 2, n - 1, and n, and the average complexity for n = 2, 3, 4, 5, 6, 7 can be found at http://im. saske.sk/~palmovsky/Prefix-free.

7. Conclusions

We studied the complexity of languages that results from the Kleene closure operation on regular and prefixfree languages. First, we proved that the *n*-state automata presented by Maslov in his 1970 paper meet the upper bound $3/4 \cdot 2^n$ on the state complexity of Kleene closure. Then, in the main result of our paper, we provided the *n*-state binary automata with *k* final states, that meet the upper bound $2^{n-1} + 2^{n-1-k}$ on the state complexity of Kleene closure. In the second part of the paper, we considered all possible values of the complexity of Kleene closure in the binary case. Our experimental results showed that there are no holes in the hierarchy up to n = 9, even in the binary case. Finally, we examined a similar problem for prefix-free languages. We showed that the state complexity of the Kleene closure of a prefix-free language with state complexity *n* may attain just three values n - 2, n - 1, and *n*.

KLEENE CLOSURE AND STATE COMPLEXITY

References

- [1] J.A. Brzozowski, In search of most complex regular languages. Internat. J. Found. Comput. Sci. 24 (2013) 691–708.
- [2] J.A. Brzozowski, E. Leiss, On equations for regular languages, finite automata, and sequential networks. *Theoret. Comput. Sci.* 10 (1980) 19–35.
- K. Čevorová, Kleene star on unary regular languages. DCFS 2013, edited by H. Jürgensen, R. Reis. In vol. 8031 of Lect. Notes Comput. Sci. Springer (2013) 277–288.
- [4] Y. Gao, L. Kari, S. Yu, State complexity of union and intersection of star on k regular languages. Theoret. Comput. Sci. 429 (2012) 98–107.
- [5] Y. Gao, N. Moreira, R. Reis, S. Yu, A review on state complexity of individual operations. Technical report, Technical Report Series DCC-2011-08, Version 1.1 Universidade do Porto. Available at www.dcc.fc.up.pt/Pubs (2016).
- [6] V. Geffert, Magic numbers in the state hierarchy of finite automata. Inform. Comput. 205 (2007) 1652–1670.
- [7] J. Hopcroft, An n log n algorithm for minimizing states in A finite automata. Technical Report STAN-CS-71-190. Computer Science Dept. (1971).
- [8] Y.S. Han, K. Salomaa, D. Wood, Operational state complexity of prefix-free regular languages. Automata, Formal Languages, and Related Topics. Institute of Informatics, University of Szeged (2009) 99–115.
- [9] K. Iwama, Y. Kambayashi, K. Takaki, Tight bounds on the number of states of DFAs that are equivalent to n-state NFAs. Theoret. Comput. Sci. 237 (2000) 485–494. Preliminary version in: 3rd International Conference on Developments in Language Theory, edited by S. Bozapalidis. Aristotle University of Thessaloniki (1997).
- [10] G. Jirásková, The Ranges of state complexities for complement, star, and reversal of regular languages. Internat. J. Found. Comput. Sci. 25 (2014) 101–124. Preliminary version: On the state complexity of complements, stars, and reversals of regular languages. DLT 2008, edited by M. Ito, M. Toyama. In vol. 5257 of Lect. Notes Comput. Sci. Springer (2008) 431–442.
- [11] G. Jirásková, Magic numbers and ternary alphabet. Int. J. Found. Comput. Sci. 22 (2011) 331–344.
- [12] G. Jirásková, M. Palmovský, J.S. Šebej, Kleene Closure on Regular and Prefix-Free Languages. CIAA 2014, edited by M. Holzer, M. Kutrib. In vol. 8587 of *Lect. Notes Comput. Sci.* Springer (2014) 226–237.
- [13] A.N. Maslov, Estimates of the number of states of finite automata. Soviet Math. Doklady 11 (1970) 1373–1375.
- [14] M. Sipser, Introduction to the theory of computation. PWS Publishing Company, Boston (1997).
- [15] M. Rabin, D. Scott, Finite automata and their decision problems. IBM Res. Develop. 3 (1959) 114–129.
- [16] J. Šebej, Reversal of regular language and state complexity. Master's thesis. P.J. Šafárik University in Košice, Slovakia (2012).
- [17] S. Yu, Regular languages, Chapter 2. In vol. I of Handbook of Formal Languages, edited by G. Rozenberg, A. Salomaa. Springer, Heidelberg (1997) 41–110.
- [18] S. Yu, Q. Zhuang, K. Salomaa, The state complexity of some basic operations on regular languages. Theoret. Comput. Sci. 125 (1994) 315–328.

Communicated by Ch. Choffrut.

Received April 21, 2016. Accepted October 14, 2016.