Kleene Star on Unary Regular Languages

Kristína Čevorová[1⋆]

Mathematical Institute, Slovak Academy of Sciences
Štefánikova 49, 814 73 Bratislava, Slovakia
cevorova@mat.savba.sk

Abstract. We study possible deterministic state complexities of languages obtained as the Kleene star of a unary language with state complexity n. We prove that for every n, depending on the parity of n, only 3 or 4 complexities from n^2-4n+6 to n^2-2n+2 are attainable. On the other hand, we show that all the complexities from 1 to n are attainable. In the end, we outline a connection to the Frobenius problem.

1 Introduction

How does the state complexity of the result of a given regular operation depend on the complexity of operands? Questions of this nature are currently objectives of many papers. Tight upper bounds for different operations both for the unary and general case have been given in [1] and others.

This determines the range of possible outcomes, but does not say anything about attainability of any particular value in this range. Two different approaches to this problem have emerged so far. Nicaud [2] studied an average case. Because even enumeration of all automata with a given state complexity is too difficult, he limited himself to basic operations on unary automata.

Another point of view was introduced by Iwama, Kambayashi and Takaki at Third Conference on Developments in Language Theory. Thier question was, whether, given any integers n and α with $n \leq \alpha \leq 2^n$, we are able to find a language with nondeterministic state complexity n and deterministic state complexity α [3]. If it is impossible, the number α is called magic for n. This problem has been solved by Jirásková for ternary alphabet [4] with a positive answer (there are no magic numbers). For a unary alphabet, a partial answer was given by Geffert [5]. He showed, that magic numbers do exist and in some sense, there are a lot of them.

Although the problem of magic numbers was originally stated for the tradeoff of nondeterminism and determinism, this idea is more universal. How does the state complexity of the language resulting from a regular operation depend on the state complexities of operands? Is the spectrum of possible outcomes continuous, or are there any gaps – magic numbers?

This question was investigated for several operations. So far it appears, that magic numbers are quite unusual phenomenon: their existence was shown only for

⋆ Research supported by grant VEGA 2/0183/11.

H. Jürgensen and R. Reis (Eds.): DCFS 2013, LNCS 8031, pp. 277–288, 2013.

determinization of unary NFAs [5] and for determinization of unary symmetric difference NFAs[6]. In the latter problem, Zijl has found necessary conditions for attainable high complexities, but it is still unknown, whether there is any large non-trivial non-magic number at all.

For the operation of Kleene Star with a growing alphabet, Jirásková has shown that each value in the range from 1 to $3/4 \cdot 2^n$ can be obtained as the state complexity of the star of an n-state DFA language [7].

This paper gives partial answer to this problem for Kleene star with unary alphabet. If the state complexity of a unary language L is n, then the state complexity of L^* is at most $(n-1)^2+1$ [1], and in the average case, it is less than a certain constant not depending on n [2].

With these results, it is not that surprising that we get two gaps of a linear length near the upper bound that are not attainable, namely, the ranges from n^2-4n+7 to n^2-3n+1 and from n^2-3n+4 to n^2-2n+1. On the other hand, the numbers n^2-3n+3 and n^2-4n+5 are attainable, and the attainability of n^2-3n+2 is determined by the parity of n. Hence we solve the problem of attainable complexities for unary star for each number in the range from n^2-4n+6 up to the known upper bound n^2-2n+2. We also show, using finite languages where possible, that values up to n can be obtained as the state complexity of the star of a unary language with state complexity n.

2 Preliminaries

In this section, we give the basic notation and definitions used in this paper.

Let $[n]=\{0,1,\ldots,n\}$, and $[c,d]$ denote the set $\{c,c+1,\ldots,d\}$ if $c \le d$, and the empty set if $c > d$.

The power set of a set A is denoted by 2^A. *The greatest common divisor* of a non-empty set S is denoted by $\gcd(S)$. The *ceiling* (*floor*) of a real number $\lceil\cdot\rceil$ ($\lfloor\cdot\rfloor$) is the smallest integer not smaller than that number (greatest integer not greater). The *state complexity* of a regular language L, $\mathrm{sc}(L)$, is the number of states of its minimal DFA.

A DFA $A=(Q,\{a\},\delta,q_0,F)$ for a unary language are uniquely given by less information than an arbitrary DFA. Identify states with numbers from $[n-1]$ via $q \sim \min\{i \mid \delta(q_0,a^i)=q\}$. Then A is unambiguously given by the number of states n, the set of final numbers F and the "loop" number $\ell=\delta(q_0,a^n)$. This allows us to freely interchange states and their ordinal numbers and justifies the notation convention used by Nicaud [2], where a unary automaton with n states, loop number ℓ and set of final states F is denoted as (n,ℓ,F). Nicaud also provided following characterization:

Theorem 1 ([2, Lemma 1]). *A unary automaton (n,ℓ,F) is minimal if and only if both conditions below are true:*

1. *its loop is minimal, and*
2. *states $n-1$ and $\ell-1$ do not have the same finality (that is, exactly one of them is final).* □

If the language of a minimal unary automaton is cofinite, then its loop has a single state, and this state is final. The state preceding the loop must be rejecting. Since this state corresponds to the longest word that is not in the language, this reasoning leads to the following proposition.

Proposition 1. *If a unary language L is cofinite, then it is regular and* $\mathrm{sc}(L) = \max\{m \mid a^m \notin L\} + 2$. □

Cofiniteness sometimes also allows us to find an upper bound on the state complexity of a language using the state complexity of another language that is accepted by some simpler automaton.

Lemma 1. *Let $0 \le \ell \le n$, and let $F_t \subseteq F_t' \subseteq [0, \ell - 1]$ and $F_\ell \subseteq F_\ell' \subseteq [\ell, n-1]$. Let $A = (n, \ell, F_t \cup F_\ell)$ be a unary automaton such that $L(A)^*$ is cofinite, and let $B = (n, \ell, F_t' \cup F_\ell')$. Then* $\mathrm{sc}(L(B)^*) \le \mathrm{sc}(L(A)^*)$. □

3 Lower Bound on Gaps in the Hierarchy of State Complexities

In this section we show that each number from 1 to n can be obtained as the state complexity of the star of an n-state unary language. Let us start with the following two technical results.

Lemma 2. *Let $A = (n, \ell, F)$ be a unary automaton and $k = \min\{F \setminus \{0\}\}$. If there exists a non-negative integer m such that $\{a^m, a^{m+1}, \ldots, a^{m+k-1}\} \subseteq L(A)^*$, then for every non-negative i, the word a^{m+i} is in $L(A)^*$.*

Proof. Every i is representable as $i = sk + r$, where r and s are non-negative integers with $r < k$. Then $a^{m+i} = a^{m+sk+r} = (a^k)^s a^{m+r}$. Since k is a final state, word a^k is in $L(A)$. By the assumption of the lemma, the word a^{m+r} is in $L(A)^*$. It follows that the word a^{m+i} is in $L(A)^*$. □

Lemma 3. *Let $\alpha \ge 7$ and $k = \lfloor \alpha/2 \rfloor$. Let $L_\alpha = \{a^k, a^{k+1}, \ldots, a^{\alpha-3}\} \cup \{a^{\alpha-1}\}$ be a finite language. Then L_α^* is cofinite and* $\mathrm{sc}(L_\alpha^*) = \alpha$.

Proof. First, notice that $a^{\alpha-2}$ is not in L_α^* since it is not in L_α and the length of any concatenation of two or more words in L_α is at least $\alpha - 1$. To prove the lemma, we only need to show that for every $i \ge 0$, the word $a^{\alpha-1+i}$ is in L_α^*. By Lemma 2, it is enough to show that $\{a^{\alpha-1}, a^{\alpha-1+1}, \ldots, a^{\alpha-1+(k-1)}\} \subseteq L_\alpha^*$.

The word $a^{\alpha-1}$ is in L_α, thus also in L_α^*. If α is even, then we have $\alpha = 2k$ and $L_\alpha = \{a^k, a^{k+1}, \ldots, a^{2k-3}\} \cup \{a^{2k-1}\}$. Therefore for $i = 1, 2, \ldots, k-2$, we have $a^{\alpha-1+i} = a^{2k-1+i} = a^k a^{k+i-1}$ which is a concatenation of words in L_α. Next $a^{\alpha-1+k-1} = a^{\alpha-3+k+1} = a^{\alpha-3} a^{k+1}$ and since $k+1 \le \alpha - 3$, this is also the concatenation of words are in L_α. The proof for an odd α is similar. □

Suppose we have a finite language L with cofinite star. We will use it to find languages, with the same state complexity of star, but greater state complexity of the language. Take any $c > \mathrm{sc}(L^*)$. Any concatenation using a^{c-2} has length at least $c-2$, but by Proposition 1, all such words already were in L^*. Therefore $\mathrm{sc}((L \cup \{a^{c-2}\})^*) = \mathrm{sc}(L^*)$ but $\mathrm{sc}(L \cup \{a^{c-2}\}) = \max\{c, n\}$.

Lemma 4. *Let* $n \geq 8$ *and* $7 \leq \alpha \leq n-1$. *There exists a unary finite language* L *such that* $\mathrm{sc}(L) = n$, *and* $\mathrm{sc}(L^*) = \alpha$.

Proof. Let L_α be the finite language given by Lemma 3. Define $L = L_\alpha \cup \{a^{n-2}\}$. Then $\mathrm{sc}(L) = n$. Since $n-2 \geq \alpha - 1$, we have $a^{n-2} \in L_\alpha^*$. Hence $\mathrm{sc}(L^*) = \alpha$. □

This is almost all we need. The following two lemmas solve missing cases.

Lemma 5. *Let* $n \geq 4$ *and* $3 \leq \alpha \leq \min\{6, n-1\}$. *There exists a unary regular language* L *such that* $\mathrm{sc}(L) = n$ *and* $\mathrm{sc}(L^*) = \alpha$.

Proof. Let $A_{n,3} = (n, 0, \{2, 3\})$, $A_{5,4} = (5, 0, \{0, 3, 4\})$ and $A_{n,4} = (n, 0, \{3, 4, 5\})$ if $n \geq 6$. Next, let $A_{n,5} = (n, 0, \{2, 5\})$ for $n \geq 6$, let $A_{7,6} = (7, 0, \{0, 3, 5, 6\})$, $A_{8,6} = (8, 7, \{6\})$ and $A_{n,6} = (n, 0, \{3, 5, 7, n-2\})$ for $n \geq 9$. Let L be the language accepted by the DFA $A_{n,\alpha}$. Then $sc(L) = n$ and $sc(L^*) = \alpha$. □

Lemma 6. *Let* $n \geq 2$ *and* $\alpha \in \{1, 2, n\}$. *There exists a unary regular language* L *such that* $\mathrm{sc}(L) = n$ *and* $\mathrm{sc}(L^*) = \alpha$.

Proof. The languages $\{a, a^{\max\{1,n-2\}}\}$ and $(a^n)^*$ satisfy the conditions of the lemma in the case of $\alpha = 1$ and $\alpha = n$, respectively.

Let $\alpha = 2$. For an even $n \geq 4$, consider the language $L = a^2(a^n)^*$ accepted by the minimal unary automaton $(n, 0, \{2\})$. For $n = 3$, let $L = a^2(a^2)^*$, and for an odd n with $n \geq 5$, let $L = a^2(a^{n-1})^* \cup \{\varepsilon\}$ be the language accepted by the minimal unary automaton $(n, 1, \{0, 2\})$. Then $sc(L) = n$ and $L^* = (a^2)^*$. □

The next theorem is a summarization of the results of this section.

Theorem 2. *For all integers* n *and* α *with* $n \geq 2$ *and* $1 \leq \alpha \leq n$, *there exists a unary regular language* L *such that* $\mathrm{sc}(L) = n$ *and* $\mathrm{sc}(L^*) = \alpha$. □

4 State Complexity of Significant Classes of Automata

In order to prove our main result, we need to find the state complexity of certain special types of automata. An useful tool for this is number theory.

Every non-negative linear combination of integers m and n will be a multiple of their common factor. Thus any number not divisible by this factor trivially does not have such a presentation. But we still may be interested in non-trivial cases of absence of such presentation. The following result is a straightforward generalization of [1, Lemma 5.1 (ii) and (iii)].

Lemma 7. *Let* m, n *be positive integers.*

a) *The largest integer divisible by* $\gcd(m, n)$ *that cannot be presented as* $mx + ny$ *for any* $x > 0, y \geq 0$ *is* $r = (\frac{m}{\gcd(m,n)} - 1)n$.
b) *The largest integer divisible by* $\gcd(m, n)$ *that cannot be presented as* $mx + ny$ *for any* $x, y \geq 0$ *is* $r = (\frac{mn}{\gcd(m,n)}) - (m + n)$. □

Now we can get the state complexity of star in some simple cases.

Theorem 3. *Let $1 \leq k < n$ and $0 \leq \ell < n$. Let L be the language accepted by a unary automaton $(n, \ell, \{k\})$.*
a) If $k < \ell$, then $\mathrm{sc}(L^*) = k$.
b) If $k \geq \ell$ and k divides $n - \ell$ then $\mathrm{sc}(L^*) = k$.
c) If $k \geq \ell$ and k does not divide $n - \ell$,
then $\mathrm{sc}(L^*) = (\frac{k}{\gcd(n-\ell,k)} - 1)(n - \ell) + \gcd(n - \ell, k) + 1$.

Proof. a) Notice that $L = \{a^k\}$. Therefore $L^* = (a^k)^*$ and $\mathrm{sc}(L^*) = k$.

b) Since $L = \{a^{k+i(n-\ell)} \mid i \geq 0\}$ and k divides $n-\ell$, the length of every word in L and L^* is divisible by k. Since $a^k \in L$, we have $L^* = (a^k)^*$ and $\mathrm{sc}(L^*) = k$.

c) Let $d = \gcd(k, n - \ell)$. The length of every non-empty word in L^* can be written as $kx + (n - \ell)y$ where $x > 0, y \geq 0$. By Lemma 7a, the maximal multiple of d unexpressable in this form is $q = (\frac{k}{d} - 1)(n - \ell)$. Since $k \nmid n - \ell$, we have $d < k$. Therefore $q > 0$ and L^* is accepted by an automaton in the form $(q + d + 1, q + 1, F)$, where $F \cap [q + 1, q + d] = \{q + d\}$, see Fig. 1. Its loop has a single final state, thus it is minimal, and the states q and $q + d$ do not have the same finality. By Theorem 1, this automaton is minimal. □

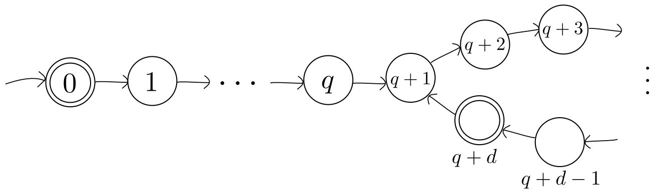

Fig. 1. The minimal automaton for star in Theorem 3c

In a similar way we can compute the state complexity of the star of languages accepted by automata with two final states, when one of them is 0.

Theorem 4. *Let $1 \leq k < n$ and $0 \leq \ell < n$. Let L be the language accepted by a unary automaton $A = (n, \ell, \{0, k\})$.*
a) If $k < \ell$, then $\mathrm{sc}(L^*) = k$.
b) If $k \geq \ell$, and k divides $n - \ell$, then $\mathrm{sc}(L^*) = k$.
c) If $k \geq \ell$, $k \nmid n - \ell$, and $\ell \neq 0$,
then $\mathrm{sc}(L^*) = (\frac{k}{\gcd(n-\ell,k)} - 1)(n - \ell) + \gcd(n - \ell, k) + 1$.
d) If $k \geq \ell$, $k \nmid n - \ell$, and $\ell = 0$, then $\mathrm{sc}(L^*) = \frac{nk}{\gcd(n,k)} - (k + n) + \gcd(n, k) + 1$. □

Until now, we did not use the construction of a DFA for star operation to get the state complexity of the resulting language. The standard construction is not difficult. To get such a DFA, we first construct an NFA for star of a given unary automaton by adding at most one new state and several transitions, and then we apply the subset construction to this NFA.

For technical reasons, we will use a slightly different construction. Let L be the language accepted by a minimal unary DFA $A = ([n-1], \{a\}, \delta, 0, F)$. Construct an NFA N from the DFA A by adding a transition on a from a state i to the state 0 whenever $\delta(i, a) \in F$. This NFA accepts L^*, except for ε if $0 \notin F$.

Suppose that $0 \notin F$. In DFA A is state 0 either unreachable by non-empty word, or reachable only from state $n-1$. It follows, that if $0 \notin F$, then in the subset automaton of N, there is no transition from any reachable state to the initial state $\{0\}$, since the only candidate for such a transition is from state $\{n-1\}$, that needs to be non-final, but by induction, all reachable states would be non-final. Thus if we mark the state $\{0\}$ in the subset automaton of N as final, we get a DFA $A' = (2^{[n-1]}, \{a\}, \delta', \{0\}, F')$ for L^*.

The DFA A' is not necessarily minimal, but it provides an upper bound on the state complexity of L^*, and this is a good starting point for a minimization.

Now we define an important notion of the set of states reached by the DFA A' after reading i symbols:

$$R_i = \delta'(\{0\}, a^i).$$

Note that $R_{i+m} = \delta'(R_i, a^m)$. Next, if $i < j$, then it does not mean that necessarily $|R_i| \leq |R_j|$. Later, we will prove this inequality with additional constraints placed on i and j. But sometimes, there are no additional requirements needed.

Lemma 8. *Let $0 \leq i < j$ and $A = (n, \ell, F)$. If $\ell = 0$ or $\ell \in F$, then $|R_i| \leq |R_j|$.*

Proof. It is sufficient to prove that $|R_i| \leq |R_{i+1}|$. If the state $n-1$ is not in R_i, then $R_{i+1} \supseteq \{q+1 \mid q \in R_i\}$, and therefore $|R_i| \leq |R_{i+1}|$.

Now assume that $n-1$ is in R_i. Since $\ell = 0$ or $\ell \in F$, the initial state 0 is in $\delta'(n-1, a)$. Therefore $R_{i+1} \supseteq \{q+1 \mid q \in R_i \setminus \{n-1\}\} \cup \{0\}$, so $|R_i| \leq |R_{i+1}|$. □

This will help us to find the complexity of more intricate type of automata.

Theorem 5. *Let $n \geq 3$ and $2 \leq k \leq n-1$. Let L be a language accepted by a unary cyclic automaton $(n, 0, [\,k,\, n-1\,])$. Then $\mathrm{sc}(L^*) = \lceil \frac{k}{n-k} \rceil k + 1$.*

Proof. First, we determine certain significant states of the DFA A' for L^*.

Let us show that for $i \in [1, \lceil \frac{k}{n-k} \rceil]$

$$R_{i \cdot k-1} = [\,n - i(n-k) - 1,\, k-1\,], \tag{1}$$

$$R_{i \cdot k} = \{0\} \cup [\,n - i(n-k),\, k\,], \tag{2}$$

and for $i \in [1, \lceil \frac{k}{n-k} \rceil - 1]$ also

$$R_{i \cdot n-1} = [\,0,\, i(n-k) - 1\,] \cup \{n-1\}. \tag{3}$$

The proof is by induction on i.

If $i = 1$, then R_{k-1} corresponds to the first $k-1$ deterministic computation steps, so $R_{k-1} = \{k-1\}$. Since the state k is final, we have $R_k = \{0, k\}$. The basis for (3) is meaningful iff $\lceil \frac{k}{n-k} \rceil - 1 \geq 1$. In that case, $n-k-1 < k$. Hence $R_{n-1} = \delta'(R_k, a^{n-1-k}) = \delta'(\{0, k\}, a^{n-1-k})$. During this part of computation, we bubble trough final states in $[k, n-1]$ and in the end, $n-1$ is reached. The transition trough each of these final states derives zero. Since $n-k-1 < k$, these zeros behaved deterministically in the subsequent computations, and finally we get $[0, n-k-1]$. Hence $R_{n-1} = [0, (n-k)-1] \cup \{n-1\}$.

Assume that $1 \leq i \leq \lceil \frac{k}{n-k} \rceil - 1$, and that (1), (2), and (3) hold. Then $R_{i \cdot n} = [0, i(n-k)]$ and is non-final, since $i(n-k) < k$. The next $(i+1)k - in - 1$ (under our assumptions, this is at least 0) steps before we reach state k are deterministic for each member state, so $R_{(i+1)\cdot k-1} = [(i+1)k - in - 1, k-1]$ and is non-final. $R_{(i+1)\cdot k} = \{0\} \cup [(i+1)k - in, k]$. In the next $(i+1)(n-k) - 1$ steps, we keep adding the initial state 0 while bubbling trough sequence of final states. In the end, this sequence reduces to $n-1$, originating from $(i+1)k - in$. Thus $R_{(i+1)\cdot n-1} = [0, (i+1)(n-k) - 1] \cup \{n-1\}$, hence (1),(2) and (3) hold.

We have $R_{\lceil \frac{k}{n-k} \rceil \cdot n} = [0, \lceil \frac{k}{n-k} \rceil (n-k)]$. It has at least $k+1$ states. Since $\ell = 0$, by Lemma 8, all its successors will have at least $k+1$ states. There are only k non-final states, so it follows that for every j with $j \geq \lceil \frac{k}{n-k} \rceil n$, the state R_j is final. Therefore, the last non-final state was $R_{\lceil \frac{k}{n-k} \rceil k-1}$. It follows that $a^{\lceil \frac{k}{n-k} \rceil k-1}$ is the longest word not accepted by the DFA A'. By Proposition 1, $\mathrm{sc}(L^*) = \lceil \frac{k}{n-k} \rceil k - 1 + 2$. $\square$

Now we will assert several dependencies between states of the DFA A', and derive an upper bound on the number of subsets reachable from its initial state, that provides estimates on the state complexity.

Lemma 9. *Let $0 \leq i < j$. Then for every non-negative integer m, we have*
a) if $R_i = R_j$, then $R_{i+m} = R_{j+m}$,
b) if $R_i \subseteq R_j$, then $R_{i+m} \subseteq R_{j+m}$.

$\square$

Proof. a) If $R_i = R_j$, then $R_{i+m} = \delta'(R_i, a^m) = \delta'(R_j, a^m) = R_{j+m}$.
b) If $R_i \subseteq R_j$, then $R_{i+m} = \delta'(R_i, a^m) \subseteq \delta'(R_i, a^m) \cup \delta'(R_j \setminus R_i, a^m) = \delta'(R_j, a^m) = R_{j+m}$.

$\square$

Corollary 1. *Let $i \geq 0$ and $k \geq 1$. If $R_i \subseteq R_{i+k}$, then the DFA A' has at most $k(n-1) + i + 1$ reachable states.*

Proof. By Lemma 9, we have a chain $R_i \subseteq R_{i+k} \subseteq R_{i+2k} \subseteq \cdots \subseteq R_{i+(n-1)k}$. Either one of these inclusions is an equality, or all of them are proper inclusions. In the first case, we have a loop in A' using less than $k(n-1) + i + 1$ subsets.

In the second case, since R_i is not empty, the set $R_{i+(n-1)k}$ has at least n elements. Hence $R_{i+(n-1)k} = \{0, 1, \ldots, n-1\}$, thus $R_{i+(n-1)k-1} \subseteq R_{i+(n-1)k}$.

By Lemma 9b, with m equal to 1, we get $R_{i+(n-1)k} \subseteq R_{i+(n-1)k+1}$, and so $R_{i+(n-1)k+1} = \{0, 1, \ldots, n-1\}$. Therefore, with an obvious inductive step, $R_{i+(n-1)k+m} = \{0, 1, \ldots, n-1\}$ for all non-negative m. It follows that the DFA A' has at most $i + (n-1)k + 1$ reachable states. $\square$

Corollary 2. *If k is a non-initial final state of A, then the DFA A' has at most $k(n-1) + 1$ reachable states.*

Proof. Let $m = \min\{q \mid q \neq 0 \text{ and } q \in F\}$. Thus $m \leq k$. Notice that we have $R_0 = \{0\} \subseteq \{0, m\} = R_m$. By Corollary 1, the DFA A' has at most $m(n-1)+1$ reachable states, and the lemma follows. □

Corollary 3. *Let $1 \leq \ell < n$, $F \setminus \{0\} \subseteq [\ell, n-1]$, and $A = (n, \ell, F)$. Then the DFA A' for the language $L(A)^*$ has at most $\ell + (n-\ell)(n-1) + 1$ reachable states.*

Proof. There is no ambiguity in a computation of NFA N after reading $\ell - 1$ symbols, thus $R_{\ell-1} = \{\ell - 1\}$. If ℓ is not final, $R_\ell = \{\ell\}$ and from definition of loop number, $\ell \in R_n$. Otherwise, if ℓ is final, $R_\ell = \{0, l\}$, but both 0 and ℓ are in R_n. Anyhow $R_\ell \subseteq R_n$ and by Corollary 1, the DFA A' has at most $\ell + (n-\ell)(n-1) + 1$ reachable states. □

5 Gaps in Complexity Hierarchy for Unary Star

In this section we present our main result. We prove that there are two gaps in the hierarchy of state complexities for unary star. The gaps are of linear length and are close to the known tight upper bound $(n-1)^2 + 1$. Since this bound follows directly from our previous observations, we provide the proof here.

Theorem 6 ([1, Theorem 5.3]). *Let $n \geq 2$ and let L be a unary regular language with $\mathrm{sc}(L) = n$. Then $\mathrm{sc}(L^*) \leq (n-1)^2 + 1$, and the bound is tight.*

Proof. If the initial state of the minimal DFA for L is a unique final state, then $L = L^*$, and $\mathrm{sc}(L^*) = n \leq (n-1)^2 + 1$.

Otherwise, there exists a final state k with $0 < k \leq n-1$. By Corollary 2, the DFA A' for L^* has at most $k(n-1) + 1 \leq (n-1)^2 + 1$ reachable states.

If $n = 2$, then the witness automaton is $(2, 0, \{0\})$. Otherwise, the witness automaton is the cyclic automaton $(n, 0, \{n-1\})$. Since $\gcd(n, n-1) = 1$, by Theorem 3c star of its language has the state complexity $(n-2)n + 1 + 1 = (n-1)^2 + 1$. □

Using previous sections, we would be able to show that various state complexities of star of n-state unary languages are attainable. Since we are interested in high complexities, the next result will be important for us.

Lemma 10. *For every $n \geq 2$, there is a unary language L such that $\mathrm{sc}(L) = n$ and $\mathrm{sc}(L^*) = (n-2)(n-1) + 1$.*

Proof. If $n = 2$, then we can take the automaton $(2, 1, \{1\})$. Otherwise, consider the unary automaton $(n, 0, \{0, n-1\})$. Since 0 and $n-1$ are subsequent states, the loop is minimal. Because $n-1$ does not divide n if $n \geq 3$, by Theorem 4d, we have $\mathrm{sc}(L^*) = n(n-1) - (n-1+n) + 1 + 1 = (n-1)(n-2) + 1$. □

The next two theorems show that, with sparse exceptions, high state complexities of star are unattainable.

Theorem 7. *Let $n \geq 3$. There is no unary language L with $\mathrm{sc}(L) = n$ and $(n-2)(n-1)+1 < \mathrm{sc}(L^*) < (n-1)^2+1$.*

Proof. We will use two estimates from the section 4 to obtain necessary conditions on minimal automaton $A = (n, \ell, F)$ recognizing some language L with $\mathrm{sc}(L^*) > (n-2)(n-1)+1$.

If 0 is the only final state, then $L = L^*$ and $\mathrm{sc}(L^*) = n < (n-2)(n-1)+2$.

Therefore, the automaton A has a final state k such that $k \geq 1$. By Corollary 2, the DFA A' for L^* has at most $k(n-1)+1$ states. We must have $k(n-1)+1 > (n-2)(n-1)+1$, and since $k < n$, the only solution is $k = n-1$. Hence $n-1$ is the only non-initial final state. It is inside the loop, so by Corollary 3, the DFA A' has at most $\ell+(n-\ell)(n-1)+1$ states. We need $\ell+(n-\ell)(n-1)+1 > (n-1)(n-2)+1$, and therefore If $n \geq 4$, then $\ell \leq 2$. If $n = 3$, then $\ell \leq 2$ as well since $[2] = \{0, 1, 2\}$.

These restrictions yield only six types of automata. The state complexities of these candidates are in Table 1. If $n \geq 3$, then none of them is in the range $[(n-2)(n-1)+2, (n-1)^2] = [n^2-3n+4, n^2-2n+1]$. □

Theorem 8. *Let $n \geq 6$. Then there is no language L such that $\mathrm{sc}(L) = n$ and $n^2-4n+6 < \mathrm{sc}(L^*) < n^2-3n+2$. Furthermore, the number n^2-3n+2 is attainable as the complexity of star if n is odd, and it is unattainable if n is even.*

Proof. Similarly as in the previous proof, we can find the restrictions on the first non-zero final state k and the loop number ℓ. By Corollary 2 we have $k \geq n-2$.

First suppose there is a non-initial final state outside the loop. Since $n-2$ is the smallest possible final state, we have only two such minimal automata: $(n, n-1, \{n-2\})$ and $(n, n-1, \{0, n-2\})$. In both cases the star is $(a^{n-2})^*$ with state complexity $n-2$.

Thus we may assume that no non-initial state outside the loop is final. Then by Corollary 3 we have $\ell \leq 3$. This yields 24 different types of automata. Tables 1 and 2 summarize complexities of stars of types with single nonzero final state. All of them could be get by direct use of Theorem 3 or 4.

Now consider automata with both states $n-1$ and $n-2$ final. Since $n-1$ and $n-2$ are final, all nonnegative linear combinations of $n-1$ and $n-2$ are in the star. If $\ell \in \{1, 2\}$, using the loop does not add anything new to the star. Since $n-1$ and $n-2$ are coprime if $n \geq 6$, by 7b, the largest integer not representable

Table 1. The candidates for L with $\mathrm{sc}(L^*) > (n-1)(n-2)+1$

type of automata	complexity	by Th.	notes
$(n, 0, \{n-1\})$	n^2-2n+2	3c	if $n \geq 3$, then $n-1 \nmid n$ and $\gcd(n, n-1) = 1$
$(n, 0, \{0, n-1\})$	n^2-3n+3	4d	
$(n, 1, \{n-1\})$	$n-1$	3b	
$(n, 1, \{0, n-1\})$		4b	not minimal
$(n, 2, \{n-1\})$	n^2-4n+6	3c	if $n \geq 4$, then $n-2 \nmid n-1$ and $\gcd(n-1, n-2) = 1$
$(n, 2, \{0, n-1\})$		4c	

Table 2. Directly computable candidates for state complexity $> n^2 - 4n + 6$

Type of automata	complexity	by Th.	notes
$(n, 3, \{n-1\})$	if n is even $n^2 - 5n + 8$	3c	for $n > 5$ is $\ell = 3 \leq n - 1$
$(n, 3, \{0, n-1\})$	if n is odd $n^2/2 - 3n + 15/2$	4c	and $n - 3 \nmid n - 1$
$(n, 0, \{n-2\})$	if n is even $n^2/2 - 2n + 3$ if n is odd $n^2 - 3n + 2$	3c	
$(n, 0, \{0, n-2\})$	if n is even $n^2/2 - 3n + 5$ if n is odd $n^2 - 4n + 4$	4d	
$(n, 1, \{n-2\})$	$n^2 - 4n + 5$	3c	not minimal
$(n, 1, \{0, n-2\})$		4c	
$(n, 2, \{n-2\})$	$n - 2$	3b	not minimal
$(n, 2, \{0, n-2\})$		4b	
$(n, 3, \{n-2\})$	$n^2 - 6n + 11$	3c	
$(n, 3, \{0, n-2\})$		4c	

as their nonnegative linear combination is $(n-1)(n-2)-(2n-3) = n^2-5n+5$. It follows that a^{n^2-5n+5} is the longest word that is not in the star. By Proposition 1, the state complexity of the star is $n^2 - 5n + 7$.

The automata with $\ell = 3$ are "supersets" of the automata in the last two rows of Table 2. Since these correspond to cofinite languages (since $n - 2$ and $n - 3$ are coprime), then by Lemma 1, their state complexity is at most $n^2 - 6n + 11$.

The automaton $A = (n, 0, \{n-2, n-1\})$ is a special case of Theorem 5 for $k = 2$ and $\mathrm{sc}(L^*) = (n-2)\lceil \frac{n-2}{2} \rceil + 1$.

If $n \geq 6$, then all obtained state complexities are at most $n^2 - 4n + 5$, except for $n^2 - 3n + 2$ if n is odd. This completes our proof. □

6 Connection to the Frobenius Problem

The problem of the state complexity of the star of a given unary language has an interesting connection to the well-known Frobenius problem. There are no results in this sections, but it shows our problem in different light.

The lengths of words in L^* forms a subsemigroup of the semigroup of natural numbers with the operation of addition. A cofinite numerical semigroup is called a numerical monoid. The maximal integer that is not a member of a given numerical monoid is called its Frobenius number and is well defined. Since numerical semigroups are finitely generated, this directly reflects our use of finite languages with cofinite star.

The problem of finding the Frobenius number given a basis of a numerical monoid is called the Frobenius problem (FP). An alternative formulation of the FP is to find the greatest natural number, that cannot be expressed as a non-negative linear combination of given natural numbers. Computing the state complexity of cofinite languages is the same problem, but with special additional constraints on coefficients. Exactly stated as follows.

Let $F_t = \{t_1, \ldots, t_i\}$, $F_\ell = \{\ell_1, \ldots, \ell_j\}$ be sets of positive integers and let r be a positive integer. Then $\tilde{f}(F_t, F_\ell, r)$ is the greatest integer not contained in the numerical semigroup

$$G(F_t, F_\ell, r) = \{\sum_{k=1}^{i} c_k t_k + \sum_{k=1}^{j} d_k \ell_k + \rho\, r \mid (c_k, d_k, \rho \geq 0) \wedge (\rho = 0 \vee \sum_{k=0}^{j} d_k > 0)\}$$

Since solving FP for basis S is equivalent with computation of $\tilde{f}(S, \varnothing, \varnothing)$, this is a generalization of FP.

On the other hand, it could be reduced to FP for the cost of a more complex basis. If we suppose, that $\ell_1 \leq \ell_k$ for all k, then one of basis of $G(F_t, F_\ell, r)$ is $\{t_1, \ldots t_i\} \cup \{l_k + mr|\ k \in [1, j], m \in [0, l_1 - 1]\}$. As a consequence, $\tilde{f}(F_t, F_\ell, r)$ is well defined iff $\gcd(F_t \cup F_\ell \cup \{r\}) = 1$.

For finite languages, we solve the classical FP. For an automaton with cofinite star $(n, \ell, \{t_1, \ldots, t_i\} \cup \{\ell_1, \ldots \ell_j\})$, where $t_k < \ell$ and $\ell_k \geq \ell$, we need to find $\tilde{f}(\{t_1, \ldots, t_i\}, \{\ell_1, \ldots \ell_j\}, \{n - \ell\})$. If the star is not cofinite, then the common divisor of all lengths is nontrivial, and we proceed similarly as in Lemma 7.

We have seen, that results in the language of FP could be translated to the language of the state complexity of unary star and vice versa. If we do this with Theorem 5, we get a generalization of Roberts's formula for FP with an arithmetic sequence as a basis[8], but only for difference 1.

7 Conclusion

We studied the state complexity of unary languages obtained as Kleene star of a language with state complexity n.

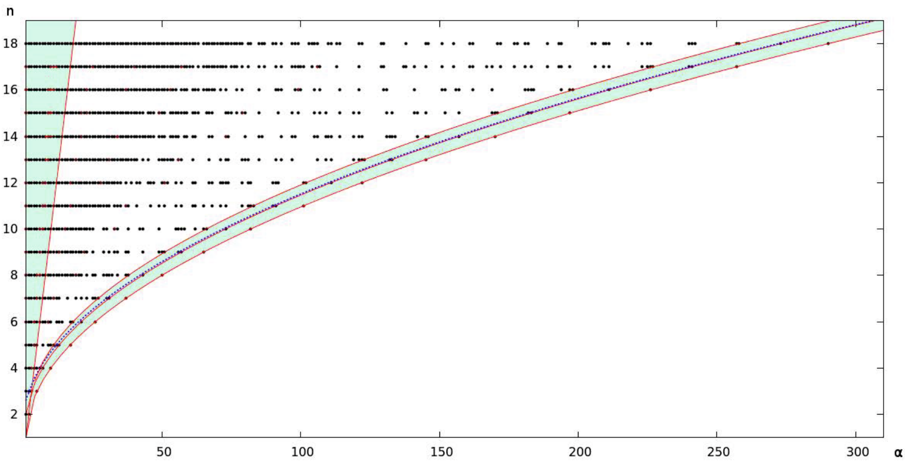

Fig. 2. Computations for $n \leq 18$

We have shown, using mostly finite languages, that we can reach any value from 1 to n as the state complexity of the star of an n-state unary language. Computations indicates, that n is not a tight lower bound on "attainable" numbers, and we assume that it could be improved significantly.

Next, we studied the range from $n^2 - 4n + 6$ to the known tight upper bound $n^2 - 2n + 2$ and we showed that the complexities $n^2 - 2n + 2$, $n^2 - 3n + 3$, and $n^2 - 4n + 6$ are reachable. Additionally, if n is odd, th complexity $n^2 - 3n + 2$ is also reachable. Our main result is, that all numbers in the studied range different from these three values are not reachable. Investigating any broadening of this interval would be probably hindered by problems with divisibility.

Fig. 2 illustrates these results on computations for unary languages with state complexity at most 18. A dot indicates an existence of a language with given properties, and its absence means that no such language exists. Lines indicate significant non-magic numbers, dashed line non-magic numbers for odd n and shaded area scope of our results.

Acknowledgement. I would like to thank to my supervisor Galina Jirásková for her guidance and to the anonymous reviewers for many valuable remarks.

References

1. Yu, S., Zhuang, Q., Salomaa, K.: The state complexities of some basic operations on regular languages. Theor. Comput. Sci. 125(2), 315–328 (1994)
2. Nicaud, C.: Average state complexity of operations on unary automata. In: Kutyłowski, M., Pacholski, L., Wierzbicki, T. (eds.) MFCS 1999. LNCS, vol. 1672, pp. 231–240. Springer, Heidelberg (1999)
3. Iwama, K., Kambayashi, Y., Takaki, K.: Tight bounds on the number of states of DFAs that are equivalent to n-state NFAs. Theor. Comput. Sci. 237(1-2), 485–494 (2000)
4. Jirásková, G.: Magic numbers and ternary alphabet. Int. J. Found. Comput. Sci. 22(2), 331–344 (2011)
5. Geffert, V.: Magic numbers in the state hierarchy of finite automata. Inf. Comput. 205(11), 1652–1670 (2007)
6. van Zijl, L.: Magic numbers for symmetric difference NFAs. Int. J. Found. Comput. Sci. 16(5), 1027–1038 (2005)
7. Jirásková, G.: On the state complexity of complements, stars, and reversals of regular languages. In: Ito, M., Toyama, M. (eds.) DLT 2008. LNCS, vol. 5257, pp. 431–442. Springer, Heidelberg (2008)
8. Roberts, J.B.: Note on linear forms. Proceedings of the American Mathematical Society 7(3), 465–469 (1956)