SQUARE ON CLOSED LANGUAGES

Kristína Čevorová

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

Abstract
The square of a language is its concatenation with itself. We study the state complexity of this square operation on suffix-, factor-, subword-, and unary-closed languages. We show that for factor-, subword-, and unary-closed languages the worst case state complexity is the same as for a concatenation of any two languages with the same state complexity, while for suffix-closed languages the result asymptotically differs by a factor of $\frac{1}{2}$.

1. Introduction

Sometimes a problem becomes less complex when you restrict the class of possible inputs. Other times, restrictions do not change the worst case complexity.

State complexity of regular operations is now a well-established discipline. Pioneered by early Russian works [8, 7], interest in the discipline was rekindled after [11] was published. The state complexity of all common operations is already known; one of the new possible avenues of research is a restriction from regular languages to subregular language families. A classic choice are unary and finite languages. In many cases they were studied along with regular languages.

A systematic study of common regular operations was carried out on all types of free [3], ideal [2] and closed [4] languages. Apart from the language-theoretic significance of these language families (which itself justifies their study), they appear to be interesting from the state complexity point of view; for many operations, the tight bounds for regular languages cannot be met by languages from these subregular classes.

None of the cited works looked at the unary operation square – a special case of concatenation – a concatenation of a language with itself. State complexity of square on regular and unary languages was given in [9]. We studied square on free, ideal and prefix-closed languages in [5]. Other types of closed languages remained open problems until now. In this paper, however, we address these gaps.

It is notable that the results on free, ideal and closed languages in [3, 2, 4] were obtained using the language-theoretic term quotient complexity instead of the standard state complexity, which

Research supported by VEGA grant 2/0084/15 and grant APVV-15-0091.

is defined by the minimal DFA. But the values coincide and quotients of a language and states of its minimal DFA have a one to one correspondence, so it is only a matter of choice. We use the state complexity wherever possible, but the proof of an upper bound for suffix-closed languages uses quotients.

The paper is organized as follows: preliminaries are in Section 2. We discuss suffix-closed languages in Section 3, while factor-, subword- and unary-closed languages are studied in Section 4. Section 5 concludes the paper.

2. Preliminaries

We assume that the reader is familiar with the basics of language and automata theory, for a reference see [10]. We recall only the essential notions here.

Cardinality of a set S is denoted by $|S|$. A partial order $\succeq$ over a poset, that is, a partially ordered set $(\mathcal{S}, \succeq)$ is a reflexive, antisymmetric and transitive binary relation over the set $\mathcal{S}$.

The *square of language* L, denoted as L^2, is a unary operation on languages that concatenates a language with itself, that is, $L^2 = L \cdot L$.

Proposition 2.1 *Let A be a DFA with a single accepting state that coincides with the initial state. Then $L(A)^2 = L(A)$.*

Let u, v, x, w be words such that $w = uvx$. We call u a *prefix*, x a *suffix* and v a *factor* of word w. If $w = u_1 v_1 u_2 v_2 \cdots u_k v_k$ for some words u_i, v_i, then the word $u = u_1 u_1 \cdots u_k$ is called a *subword* of w. For unary languages, all these terms coincide.

L is *xfix-closed* if whenever u is an xfix of v and $v \in L$, then also $u \in L$. Moreover, if L is unary, we call it *unary-closed.*

An *xfix-closure* is an operator $\mathrm{cl}_{xfix}(\cdot)$ on languages and the language $\mathrm{cl}_{xfix}(L)$ is the minimal xfix-closed language containing L.

The *state complexity of a regular language* L is a number $\mathrm{sc}(L)$ that represents the number of states of the minimal DFA recognizing L.

3. Suffix-Closed Languages

How to discern that a DFA recognizes a suffix-closed language? The following two propositions provide an outline of a deciding algorithm.

Proposition 3.1 *[6, Theorem 10] Let $A = (Q, \Sigma, \delta, q_0, F)$ be a DFA without unreachable states. Then the NFA $(Q, \Sigma, \delta, Q, F)$ recognizes the suffix-closure of $L(A)$.*

Proposition 3.2 *A language is suffix-closed if and only if it is equal to its suffix-closure.*

Alter the DFA by marking all states as initial, determinize the resulting NFA and check whether you obtained a DFA equivalent with the original. Both determinization and minimization makes this deciding algorithm impractical to use on classes of DFAs, so a sufficient structural condition guaranteeing that a DFA recognizes a suffix-closed language, is worthwhile. We introduce a condition that requires a special sortability of states.

Lemma 3.3 *Let $A = (\{0, 1, \dots, n-1\}, \Sigma, \delta, 0, F)$ be an n-state DFA. If*

1. *$F = \{0, 1, \dots, k\}$ for some integer k and*
2. *every letter is a nondecreasing function, that is, if $x < y$, then $\delta(x, a) \leq \delta(y, a)$ for every letter $a \in \Sigma$,*

then $L(A)$ is a suffix-closed language.

Proof.

The key idea is, that the nondecreasing property of letters in A could be inductively generalized to words, that is, if one takes two computations on the same word from two different states, then the ending states will be in the same ordering as the starting states.

We will use the characterization of suffix-closed languages from Proposition 3.2 and show that $L(A) = \mathrm{cl}_{suffix}(L(A))$. Consider the DFA A and an NFA A' recognizing $\mathrm{cl}_{suffix}(L(A))$ as in Proposition 3.1. The only difference between them is that the NFA has all states initial. Therefore $L(A') \supseteq L(A)$.

Consider an accepting computation of the NFA A' on word w. The only nondeterministic choice is the selection of the first state. This computation is accepting, therefore ends in some accepting state q and $q \leq k$. A computation from the lowest state 0 cannot end in state higher than q, therefore the computation on w in A is also accepting. Hence also $L(A') \subseteq L(A)$. □

Upper bound

To obtain an upper bound on the state complexity of square of a suffix-closed language, we will use quotients of a language and the quotient complexity of a language that always has the same numerical value as the state complexity of that language.

Let L be a language and w any word (not neccessarily from L). Then the quotient of L by w is the language $L_w = \{u \in \Sigma^* \mid wu \in L\}$. The number of distinct quotients of L is called the quotient complexity of L and is denoted as $\mathcal{K}(L)$. A quotient L_w is called accepting if $\varepsilon \in L_w$; there is a one to one correspondence between quotients of a language and states of its minimal DFA, where accepting quotients correspond to accepting states, hence the name. This correspondence also ensures that the number of distinct quotients of a language – the quotient complexity of a language – is equal to its state complexity. For a more detailed explanation of quotients and quotient complexity, see [1].

Let L be a suffix-closed language with $\mathcal{K}(L) = n$. Since ε is a suffix of all words, $\varepsilon \in L$ if L is non-empty. It follows that L_ε is an accepting quotient unless L is empty. Moreover, $L \subseteq L^2$.

Remark 3.4 *If L_ε is the only accepting quotient of L, then the minimal DFA accepting L has only one accepting state coinciding with the initial state and by Proposition 2.1 we have that $L^2 = L$, thus $\mathcal{K}(L^2) = n$.*

The hierarchical structure of quotients of a suffix-closed language is neatly organized; Remark 2 in [4] states the following relationship between quotients: if v is a suffix of w, then $L_w \subseteq L_v$; in particular $L_w \subseteq L_\varepsilon$ for every quotient L_w.

This enables us to study the poset of quotients of a suffix-closed language via the defining words. Before doing so, we interrupt with a poset-related result that will be used later.

Lemma 3.5 *Let $(\mathcal{S}, \succeq)$ be a partially ordered set with n elements. Let $\alpha : \mathcal{S} \longrightarrow \mathbb{N}$ be a function counting the number of elements that are above the given element, formally defined as $\alpha(s) = |\{r \in \mathcal{S} \mid r \succeq s\}|$. Then*

$$\sum_{s\in\mathcal{S}} \alpha(s) \leq \frac{1}{2}(n^2 + n).$$

Proof. We will estimate the value $\alpha(s)$ for every element $s \in \mathcal{S}$. Take any minimal element m of $\mathcal{S}$. Clearly $\alpha(m) \leq n$. Because m is minimal, we can remove it from the poset without changing the value of function α restricted to the domain $\mathcal{S} \setminus \{m\}$.

This removing procedure can be iterated until the poset is empty and note that the i-th removed element has value of α of at most $n - i + 1$. Therefore the sum over all elements of $\mathcal{S}$ is at most $n + (n - 1) + \cdots + 2 + 1 = \frac{1}{2}(n^2 + n)$. □

Now we are ready to prove the key result of this section. It heavily relies on the work on the product of suffix-closed languages in [4].

Theorem 3.6 *Let $n \geq 3$ and L be a suffix-closed language with the quotient complexity n. Then $\mathcal{K}(L^2) \leq \frac{1}{2}(n^2 + n) - 1$.*

Proof. In the proof of [4][Theorem 3, case 2.] the authors derive an equation for a quotient of concatenation of suffix-closed languages. Square is a concatenation of a language with itself, so after substituting we obtain the following equation:

$$(L^2)_w = L_w L \cup L_x \text{ for some suffix } x \text{ of } w.$$

How many distinct quotients can we get? We fix a quotient L_w and identify all possible pairings with quotients L_x. Consider the partially ordered set of quotients of L with the set inclusion. Recall that if x is a suffix of w, then $L_w \subseteq L_x$. Therefore all pairable L_x are above L_w in this poset. We are interested in the sum for all quotients and so our question reduces to Lemma 3.5 with the set of all quotients and $\supseteq$ as the partial order. The overall sum is $\frac{1}{2}(n^2 + n)$.

This upper bound can be slightly improved by taking the quotient acceptingness into account. Recall that the quotient L_ε of a non-empty suffix-closed language L is always accepting. Hence, if L has only one accepting quotient, it is the quotient L_ε. By Remark 3.4, for any language L with the only accepting quotient L_ε holds $\mathrm{sc}(L^2) = n$. What if L has more than one accepting quotient?

If a quotient L_w is accepting, then by [4] we have that $(L^2)_w = L_w L$, thus L_w is associated with only one quotient of L^2. And in Lemma 3.5, for every quotient except L_ε we counted at least two quotients $\succeq$ than it – the quotient itself and L_ε. Thus for languages with more than one accepting quotients we counted at least one extra quotient. After subtracting 1, we obtain an upper bound $\frac{1}{2}(n^2+n)-1$ for such languages. Since for $n \geq 2$ holds that $n \leq \frac{1}{2}(n^2+n)-1$, we obtain also a universal upper bound $\frac{1}{2}(n^2+n)-1$. □

Lower bound

In the previous section, we obtained the upper bound $\frac{1}{2}(n^2+n)-1$ on the state complexity of square of a suffix-closed language. To show that it is tight, we will construct a witness DFA.

Construction 3.7 *Let $n \geq 4$. Let $A = (\{0,1,\dots,n-1\},\{a,b,c\},\delta,\{0\},\{0,1\})$ be a DFA with the following transition function δ:*

$$\delta(i,a) = \begin{cases} i, & \text{if } i \in \{0,1,n-1\};\\ i-1, & \text{if } i \in \{2,\dots,n-3\};\\ i+1, & \text{if } i = n-2. \end{cases}$$

$$\delta(i,b) = \begin{cases} i, & \text{if } i = 0;\\ i-1, & \text{otherwise.} \end{cases}$$

$$\delta(i,c) = \begin{cases} i+1, & \text{if } i \in \{0,1,\dots,n-3\};\\ i, & \text{otherwise.} \end{cases}$$

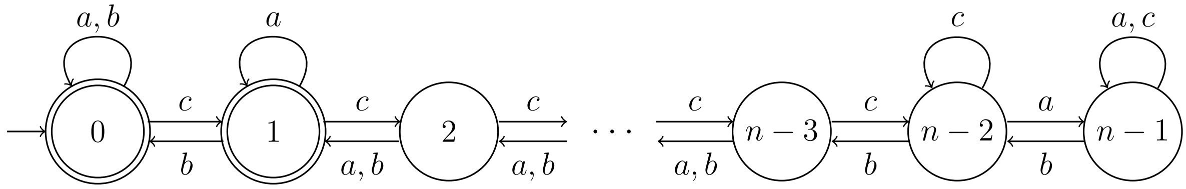

Figure 1: A ternary witness for optimality of the upper bound $\frac{1}{2}(n^2+n)-1$ on suffix-closed languages.

To show that DFA A from Construction 3.7 is indeed a witness, we need to verify its suffix-closedness and that the state complexity of $L(A)^2$ is equal to $\frac{1}{2}(n^2+n)-1$. There is no need to check the minimality of A because by Theorem 3.6, a square of a suffix-closed language with state complexity lower than n has state complexity at most $\frac{1}{2}((n-1)^2+(n-1))-1$.
Note that A is by Lemma 3.3 suffix-closed, because accepting states 0 and 1 form a contiguous

interval and the function $\delta(\cdot, x)$ is non-decreasing for all letters x.
The following construction introduces a DFA recognizing $L(A)^2$.

Construction 3.8 *To construct the DFA A^2 accepting the language $L(A)^2$, we first construct an NFA for $L(A)^2$ containing two structural copies of A as in Figure 2. Then we determinize it by the standard Rabin-Scott subset construction.*

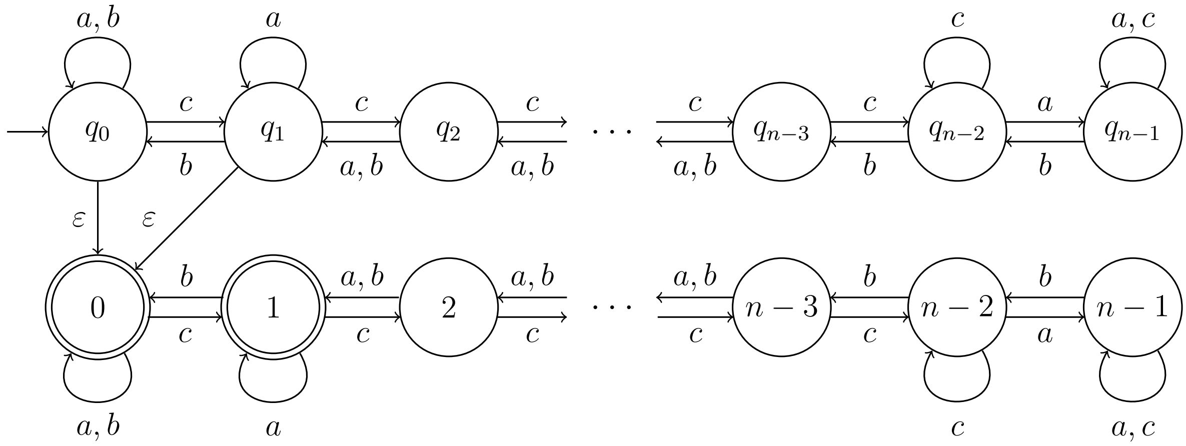

Figure 2: An NFA for square of the suffix-closed ternary witness.

Technically, the DFA A^2 has 2^{2n} states, but not all of them are necessarily reachable. We will show that at least $\frac{1}{2}(n^2 + n) - 1$ of them are.

Lemma 3.9 *In the DFA A^2 obtained from DFA A from Construction 3.7 by Construction 3.8, all states from the sets $\{\{q_i, m, i\} \mid 0 \le m < i \le n-1\}$ and $\{\{q_i, i\} \mid 0 \le i \le n-1\}$ except $\{q_1, 1\}$ are reachable.*

Proof. We will divide the states into four types and show their reachability:

- $\{q_i, 0, i\}$:

$$\{q_0, 0\} \xrightarrow{c^{n-2}} \{q_{n-2}, n-3, n-2\} \xrightarrow{a} \{q_{n-1}, n-4, n-1\} \xrightarrow{a^{n-5}} \cdots$$
$$\cdots \xrightarrow{a^{n-5}} \{q_{n-1}, 1, n-1\} \xrightarrow{b} \{q_{n-2}, 0, n-2\} \xrightarrow{b^{n-2-i}} \{q_i, 0, i\}$$

  The scheme above does not show reachability of state $\{q_{n-1}, 0, n-1\}$, which can be reached as
$$\{q_{n-2}, 0, n-2\} \xrightarrow{a} \{q_{n-1}, 0, n-1\}.$$

- $\{q_{i+m}, i, i+m\}$ for $1 \le m \le n-2-i$: Note that we do not show reachability of states containing the state q_{n-1} here.

$$\{q_m, 0, m\} \xrightarrow{c^i} \{q_{i+m}, i, i+m\}$$

- $\{q_{n-1}, i, n-1\}$:

$$\{q_{n-1}, 0, n-1\} \xrightarrow{c^i} \{q_{n-1}, i, n-1\}$$

- $\{q_i, i\}$:
$$\{q_{n-1}, n-2, n-1\} \xrightarrow{a} \{q_{n-1}, n-1\} \xrightarrow{b^{n-1-i}} \{q_i, i\} \qquad \square$$

In order to obtain the lower bound, we need to show that all states reached in the previous lemma are also distinguishable.

Lemma 3.10 *Let A^2 be the DFA from the previous lemma. States of A^2 from sets $\{\{q_i, m, i\} \mid 0 \leq m < i \leq n-1\}$ and $\{\{q_i, i\} \mid 0 \leq i \leq n-1\}$ can be pairwise distinguished.*

Proof. Whenever we write $\{q_i, m, i\}$ in this proof, we also admit the case $m = i$ and this is in fact the state $\{q_i, i\}$. We will distinguish states $\{q_i, m, i\}$ and $\{q_j, n, j\}$. The proof splits into two cases.

First, suppose that $m \neq n$ so without loss of generality $m < n$. If $m = 0$ and $n \geq 2$, the former is accepting and the latter is not, so they are not equivalent. If $m = 0$ and $n = 1$, the letter c distinguishes them, because one of the resulting states contains the state 1 and is thus accepting, while the second has the lowest state of at least 2 and is non-accepting. The last subcase is $1 \leq m < n$. After reading b^{m-1}, the computation from one of the states leads to the accepting state $\{q_{i-m+1}, 1, i-m+1\}$, while from the other to the state $\{q_{j-m+1}, n-m+1, j-m+1\}$, which is non-accepting because $n-m+1$ is at least 2 and $j-m+1$ is even more.

Now suppose that $n = m$. Therefore $i \neq j$ and we may suppose $i < j$. After reading the word b^i we end in states $\{q_0, 0\}$ and $\{q_{j-i}, 0, j-i\}$. Then we read the word cc; the former ends in state $\{q_2, 1, 2\}$ and the latter in state $\{q_t, 2, t\}$ for some $t \geq 2$. The former is accepting, while the latter is not. $\square$

Previous two lemmata help us to provide a lower bound on state complexity of square on suffix-free languages, which is the key result of this section.

Corollary 3.11 *State complexity of the language recognized by the DFA A^2 constructed from DFA A from Construction 3.7 by Construction 3.8 is $\frac{1}{2}(n^2+n)-1$.*

Proof. By Lemma 3.9 the DFA A^2 has at least $\frac{1}{2}(n^2+n)-1$ states and Lemma 3.10 shows that all of them are pairwise inequivalent. Therefore $\mathrm{sc}(L(A^2)) \geq \frac{1}{2}(n^2+n)-1$. But since DFA A^2 recognizes a square of a suffix-closed language recognized by an n-state DFA, by Theorem 3.6 we also have $\mathrm{sc}(L(A^2)) \leq \frac{1}{2}(n^2+n)-1$. $\square$

We just obtained a lower bound. Compare it with the upper bound we already had – they are identical. So our upper bound is tight and together we get:

Theorem 3.12 *For $n \geq 2$, the state complexity of the square operation on the class of suffix-closed languages is $\frac{1}{2}(n^2+n)-1$, and for $n = 1$ it is 1.*

Proof. Theorem 3.6 shows the upper bound on complexity $\frac{1}{2}(n^2+n)-1$. The DFA A from Construction 3.7 therefore could not be non-minimal, showing that this bound is tight for $n \geq 4$. Languages $\{w \in \{a,b\}^* \mid bb \text{ is not a subword of } w\}$, Σ^* and $\{\varepsilon\}$ are the witnesses for $n = 3$, 2 and 1 respectively. $\square$

4. Subword-, Factor- and Unary-Closed Languages

The state complexity of concatenation is $m + n - 1$ both on subword-closed and factor-closed languages. This gives an upper bound $2n-1$ on the state complexity of square on these classes. We will find a family of subword-closed languages that proves the tightness of this bound for both types, since a subword-closed language is also factor-closed.

Lemma 4.1 *For every* $n \geq 3$ *there exists a binary subword-closed language* L *such that* $\mathrm{sc}(L) = n$ *and* $\mathrm{sc}(L^2) = 2n - 1$.

Proof. Fix n and consider the language L of all words that can be divided into two partitions $w = w'a^i$ where every word w' contains at most $n-3$ letters a and i is an arbitrary non-negative integer. A DFA A recognizing the language L is depicted in Figure 3.

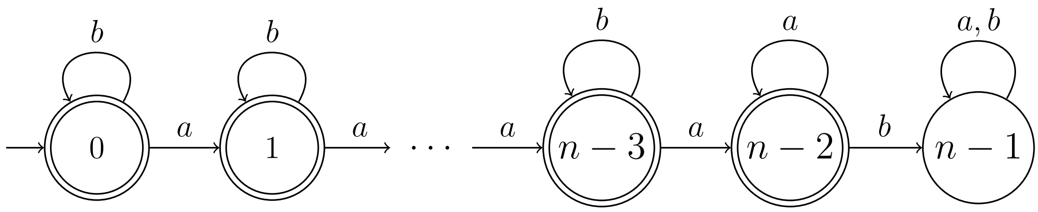

Figure 3: A binary witness for optimality of the upper bound $2n - 1$ on subword- and factor-closed languages. States from 0 to $n-3$ count the number of already read a, state $n-2$ enables unlimited reading of a and $n-1$ is the dead state.

Any letter can be omitted from any of the partitions – number of occurrences of the letter a in the partition w' can not increase and a^{i-1} is also a plausible second partition. Therefore the language L is subword-closed. Since we have a semantical description of the language L, we can construct a DFA A^2 as shown in Figure 4, recognizing the language L^2 directly.

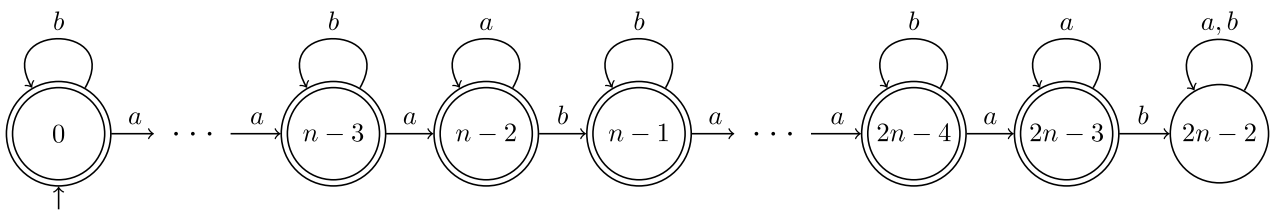

Figure 4: The DFA A^2 for the square of the language of the DFA in Figure 3.

An outline why the DFA A^2 in Figure 4 indeed recognizes L^2:

First we show that $L(A^2) \subseteq L^2$. Consider any accepting computation of A^2. Since the sequence of current states is nondecreasing, we can divide the computation into two parts, first containing only states from $\{0, 1, \ldots, n-2\}$ and the second only from $\{n-1, n, \ldots, 2n-3\}$. Each part corresponds to an accepting computation of A.

To prove equality, we also show the reverse inclusion $L(A^2) \supseteq L^2$. To prove this by a contradiction, suppose there is a word w that is not accepted by A^2 but could be divided into two words

w_1, w_2 from L such that $w = w_1w_2$. Consider the non-accepting computation of A^2 on w – it ends in the state $2n-2$. Let q be the state reached after reading of w_1 in A^2. Either $q > n-2$ and then w_1 cannot be accepted by A, or $q \le n-2$ and then w_2 cannot be accepted by A. That is a contradiction with assumption that both w_1 and w_2 are from L.

To show that A^2 is minimal, consider the word $d = a^{n-2}ba^{n-2}b$ and let d_i be the suffix of d containing the last $2n-2-i$ letters. Let i and j be two distinct states of A^2, without loss of generality $i < j$. Word d_j distinguishes these states because computation on d_j from state i ends in some accepting state, while computation from state j ends in the sole non-accepting state $2n-2$. Therefore the DFA A^2 is minimal and $\mathrm{sc}(L^2) = 2n-1$. Since $\mathrm{sc}(L^2) \ge 2(n-1)-2$, the subword-closed language L cannot be recognized by fewer than n states, therefore $\mathrm{sc}(L) = n$.

The language L is therefore a subword-closed language with state complexity n and state complexity of square of L is $2n-1$. □

For unary languages, terms suffix, factor, and subword coincide. The only *unary-closed* language with state complexity n is $\{a^i \mid i \le n-2\}$. The square of this language is $\{a^i \mid i \le 2n-4\}$ and its state complexity is $2n-2$.

5. Conclusions

We studied the state complexity of the operation square on various types of closed languages. Results on concatenation in [4] provide an instant upper bound but except for unary languages, the witnesses of tightness provided by the authors cannot be reused for concatenation of a language with itself. Yet poorer upper bound can be obtained from the results on square on regular languages in [9].

We showed that the bound from concatenation is tight for factor-, subword- and unary-closed languages. For suffix-closed languages the bound is only asymptotically tight with factor $\frac{1}{2}$. We provided a new upper bound and proved its optimality on a ternary alphabet. Tightness on a binary alphabet is an open question, but computations suggest that this upper bound cannot be attained by a binary language. Our results solve the open problems in [5]. We also obtained a structural sufficient condition on DFAs recognizing suffix-closed languages.

		square	$\lvert\sum\rvert$	concatenation	$\lvert\sum\rvert$
closed	unary	$\mathbf{2n-2}$		$m+n-2$	
	suffix	$\mathbf{\frac{1}{2}(n^2+n)-1}$	3	$(m-1)n+1$	3
	prefix	$(n+4)2^{n-3}-1$	2	$(m+1)2^{n-1}$	3
	factor, subword	$\mathbf{2n-1}$	2	$m+n-1$	2
regular	unary	$2n-1$		mn	if $(m,n)=1$
	general	$n2^n-2^{n-1}$	2	$m2^n-2^{n-1}$	2

Table 1: Comparison of state complexity of square and concatenation on closed and regular languages.

References

[1] J. A. BRZOZOWSKI, Quotient Complexity of Regular Languages. *Journal of Automata, Languages and Combinatorics* 15 (2010) 1/2, 71–89.

[2] J. A. BRZOZOWSKI, G. JIRÁSKOVÁ, B. LI, Quotient Complexity of Ideal Languages. *Theoretical Computer Science* 470 (2013), 36–52.
`http://dx.doi.org/10.1016/j.tcs.2012.10.055`

[3] J. A. BRZOZOWSKI, G. JIRÁSKOVÁ, B. LI, J. SMITH, Quotient Complexity of Bifix-, Factor-, and Subword-Free Regular Languages. In: P. DÖMÖSI, S. IVÁN (eds.), *AFL*. AFL 2011, 2011, 123–137.
`http://dblp.uni-trier.de/db/conf/afl/afl2011.html#BrzozowskiJLS11`

[4] J. A. BRZOZOWSKI, G. JIRÁSKOVÁ, C. ZOU, Quotient Complexity of Closed Languages. *Theory of Computing Systems* 54 (2014) 2, 277–292.
`http://dx.doi.org/10.1007/s00224-013-9515-7`

[5] K. ČEVOROVÁ, Square on Ideal, Closed and Free Languages. In: J. SHALLIT, A. OKHOTIN (eds.), *Descriptional Complexity of Formal Systems - 17th International Workshop, DCFS 2015, Waterloo, ON, Canada, June 25-27, 2015. Proceedings*. Lecture Notes in Computer Science 9118, Springer, 2015, 70–80.
`http://dx.doi.org/10.1007/978-3-319-19225-3_6`

[6] J.-Y. KAO, N. RAMPERSAD, J. SHALLIT, On NFAs Where All States are Final, Initial, or Both. *Theoretical Computer Science* 410 (2009) 4749, 5010 – 5021.
`http://www.sciencedirect.com/science/article/pii/S0304397509005477`

[7] A. N. MASLOV, Estimates of the Number of States of Finite Automata. *Doklady Akademii Nauk SSSR* 194 (1970), 1266–1268. English translation in Soviet Mathematics Doklady 11 (1970) 1373–1375.

[8] B. G. MIRKIN, On Dual Automata. *Kibernetica* 2 (1966) 1, 7–10. English translation in Cybernetics 2, 69 (1970).

[9] N. RAMPERSAD, The State Complexity of L^2 and L^k. *Information Processing Letters* 98 (2006) 6, 231–234.

[10] M. SIPSER, *Introduction to the Theory of Computation*. Second edition, Course Technology, 2006.

[11] S. YU, Q. ZHUANG, K. SALOMAA, The State Complexities of Some Basic Operations on Regular Languages. *Theoretical Computer Science* 125 (1994) 2, 315 – 328.
`http://www.sciencedirect.com/science/article/pii/030439759200011F`