The Range of State Complexities of Languages Resulting from the Cut Operation

Markus Holzer[1] and Michal Hospodár[2](✉)

[1] Institut für Informatik, Universität Giessen, Arndtstr. 2, 35392 Giessen, Germany
holzer@informatik.uni-giessen.de
[2] Mathematical Institute, Slovak Academy of Sciences, Grešákova 6, 040 01 Košice, Slovakia
hosmich@gmail.com

Abstract. We investigate the state complexity of languages resulting from the cut operation of two regular languages represented by minimal deterministic finite automata with m and n states. We show that the entire range of complexities, up to the known upper bound, can be produced in the case when the input alphabet has at least two symbols. Moreover, we prove that in the unary case, only complexities up to $2m-1$ and between n and $m+n-2$ can be produced, while if $2m \leq n-1$, then the complexities from $2m$ up to $n-1$ cannot be produced.

1 Introduction

It is well known that for every n-state nondeterministic finite automaton (NFA), there exists a language-equivalent deterministic finite automaton (DFA) with at most 2^n states [21]. This bound is tight in the sense that for an arbitrary integer n there is always some n-state NFA which cannot be simulated by any DFA with less than 2^n states [17–19,23].

Nearly two decades ago a very fundamental question on determinization was raised by Iwama, Kambayashi, and Takaki [9]: does there always exist a minimal n-state NFA whose equivalent minimal DFA has α states for all n and α with $n \leq \alpha \leq 2^n$? Iwama, Matsuura, and Paterson [10] called a number α in the range from n to 2^n *magic* if no minimal n-state NFA has an equivalent minimal α-state DFA. The simple question whether for every n no number is magic turned out to be harder than expected. In a series of papers, non-magic (attainable) numbers were identified [6,11,12] until the problem was solved in [14] showing that for ternary languages *no* magic numbers exist. On the contrary, Geffert [5] proved that most of the numbers in the range from n up to $F(n) + n^2$, where $F(n)$

M. Hospodár—Research supported by VEGA grant 2/0132/19 and by grant APVV-15-0091. This work was conducted during a research visit at the Institut für Informatik, Universität Giessen, Germany, funded by the DAAD short-term grant ID 57314022.

C. Martín-Vide et al. (Eds.): LATA 2019, LNCS 11417, pp. 190–202, 2019.
https://doi.org/10.1007/978-3-030-13435-8_14

is the Landau function, is not attainable as the state complexity of a language accepted by a minimal unary n-state NFA. However, his proof is existential, and no specific value is known to be unattainable. For binary languages, the original problem from [9] is still open.

The idea behind the magic number problem is not limited to the determinization of NFAs. In fact every (regularity preserving) formal language operation can be used to define a magic number problem for the operation in question. For instance, consider the intersection operation on languages. Let A and B be minimal finite automata with m and n states, respectively. Then the size of the minimal automaton for the intersection of $L(A)$ and $L(B)$ is between 1 and mn. The value one is induced by the intersection of disjoint languages and the value mn by the standard cross-product construction for the intersection operation. Thus, in a similar way as for the determinization, one may now ask, whether every α within the range between 1 and mn can be attained by the size of minimal automaton for intersection of languages given by two minimal automata with m and n states, respectively? In other words, is the outcome of the intersection operation in terms of the number of states contiguous or are there any gaps, hence magic numbers? In [8] it was shown that for the intersection on DFAs *no* number from 1 up to mn is magic—this already holds for binary automata. Besides intersection, also other formal language operations were investigated from the "magic number" perspective. It turned out that magic numbers are quite rare, and most of them occur in the unary case. For example, Čevorová [2] studied the complexity of languages resulting from the Kleene star operation in the unary case. In such a case, the known upper bound is $(n-1)^2+1$ [24]. She proved that the values from 1 to n, as well as the values n^2-2n+2 and n^2-3n+3, are attainable, while the value n^2-3n+2 is attainable if n is odd and it is not attainable otherwise. Moreover, she showed that all the values from n^2-3n+4 up to n^2-2n+1 and from n^2-4n+7 up to n^2-3n+1 cannot be attained by the state complexity of the Kleene star of any language accepted by minimal unary DFA with n states. The magic number problem was also examined for concatenation [13,16], square [3], star on general alphabet [15], and reversal [22].

We contribute to the list of magic number problems for formal language operations by studying the cut operation. The cut operation was introduced in [1] as a machine implementation of "concatenation" on UNIX text processors which behaves greedy-like in its left term of concatenation. Tight upper bounds for the state complexity of the cut and iterated cut operations on DFAs were obtained in [4]. While the state complexity of concatenation is growing linearly with the first parameter (the number of states of the left automaton) and exponentially with the second parameter (the number of states of the right automaton), the state complexity of the cut operation is only linearly growing with both parameters. In the general case, the known tight upper bound is given by the function $f(m,n)$ such that $f(m,1)=m$ and $f(m,n)=(m-1)n+m$ if $n \geq 2$. In the unary case, the known tight upper bound is given by the function

$f_1(m,n)$ such that $f_1(1,n) = 1$, $f_1(m,1) = m$, $f_1(m,n) = 2m-1$ if $m,n \geq 2$ and $m \geq n$, and finally let $f_1(m,n) = m+n-2$ if $m,n \geq 2$ and $m < n$ [4].

In this paper, we show for every value from 1 up to $f_1(m,n)$ whether or not it can be attained by the state complexity of the cut of two languages accepted by minimal unary DFAs with m and n states. We show that only complexities up to $2m-1$ and between n and $m+n-2$ can be attained, while complexities from $2m$ up to $n-1$ turn out to be magic. To get these results, the tail-loop structure of minimal unary DFAs is very valuable in the proofs.

On the other hand, we show that the entire range of complexities, up to the known upper bound $f(m,n)$, can be produced by the cut operation on minimal DFAs with m and n states, respectively, in case when the input alphabet consists of at least two symbols. The proof of this result resembles some ideas used in [8] for the magic number problem of the intersection and union operations on DFAs.

To the best of our knowledge, this is the first operation where for every alphabet, every value in the range of possible complexities is known to be either attainable or not, and not all values are attainable in the unary case. However, all values are attainable in every other alphabet size. Hence, the magic number problem for the cut operation is completely solved in this paper.

2 Preliminaries

We recall some definitions on finite automata as contained in [7]. Let Σ^* denote the set of all words over a finite alphabet Σ. The *empty word* is the word with length zero. If u, v, w are words over Σ such that $w = uv$, then u is a *prefix* of w. Further, we denote the set $\{i, i+1, \ldots, j\}$ by $[i,j]$ if i and j are integers.

A *deterministic finite automaton* (DFA) is a quintuple $A = (Q, \Sigma, \delta, s, F)$ where Q is a finite nonempty set of *states*, Σ is a finite nonempty set of *input symbols*, $s \in Q$ is the *initial* state, $F \subseteq Q$ is the set of *final* (or *accepting*) states, and $\delta \colon Q \times \Sigma \to Q$ is the *transition function* which can be extended to the domain $Q \times \Sigma^*$ in the natural way. The *language accepted* (or *recognized*) by the DFA A is defined as $L(A) = \{\, w \in \Sigma^* \mid \delta(s,w) \in F \}$.

Two DFAs A and B are *equivalent* if they accept the same language, that is, if $L(A) = L(B)$. An automaton is *minimal* if it admits no smaller equivalent automaton with respect to the number of states. For DFAs this property can be verified by showing that all states are reachable from the initial state and all states are pairwise distinguishable. It is well known that every regular language has a unique, up to isomorphism, minimal DFA.

The *state complexity* of a regular language is the number of states in the minimal DFA recognizing this language.

In [1] the *cut operation* on languages K and L, denoted by $K \,!\, L$, is defined as

$$K \,!\, L = \{\, uv \mid u \in K, v \in L, \text{ and } uv' \notin K \text{ for every nonempty prefix } v' \text{ of } v\}.$$

The above defined cut operation preserves regularity as shown in [1]. Since we are interested in the descriptional complexity of this operation we briefly recall the construction of a DFA for the cut operation; we slightly deviate from the presentation of the construction given in [4].

Let $A = (Q_A, \Sigma, \delta_A, s_A, F_A)$ and $B = (Q_B, \Sigma, \delta_B, s_B, F_B)$ be two DFAs. Let $\perp \notin Q_B$. Define the cut automaton $A\,!\,B = (Q, \Sigma, \delta, s, F)$ with the state set $Q = (Q_A \times \{\perp\}) \cup (Q_A \times Q_B)$, the initial state $s = (s_A, \perp)$ if the empty word is not in $L(A)$ and $s = (s_A, s_B)$ otherwise, the set of final states $F = Q_A \times F_B$, and for each state (p, q) in Q and each input a in Σ we have

$$\delta((p,\perp),a) = \begin{cases} (\delta_A(p,a),\perp), & \text{if}\, \delta_A(p,a) \notin F_A; \\ (\delta_A(p,a), s_B), & \text{otherwise}; \end{cases}$$

and

$$\delta((p,q),a) = \begin{cases} (\delta_A(p,a),\delta_B(q,a)), & \text{if}\, \delta_A(p,a) \notin F_A; \\ (\delta_A(p,a), s_B), & \text{otherwise.} \end{cases}$$

Then $L(A\,!\,B) = L(A)\,!\,L(B)$.

In [4], the following functions were introduced.

$$f(m,n) = \begin{cases} m, & \text{if } n = 1; \\ (m-1)n + m, & \text{if } n \geq 2 \end{cases} \qquad (1)$$

and

$$f_1(m,n) = \begin{cases} 1, & \text{if } m = 1; \\ m, & \text{if } m \geq 2 \text{ and } n = 1; \\ 2m-1, & \text{if } m,n \geq 2 \text{ and } m \geq n; \\ m+n-2, & \text{if } m,n \geq 2 \text{ and } m < n \end{cases} \qquad (2)$$

It was proven in [4, Theorems 3.1 and 3.2] that if A and B are DFAs with m and n states, respectively, then $f(m,n)$ states, resp. $f_1(m,n)$ states if A and B are unary, are sufficient and necessary in the worst case for any DFA accepting the language $L(A)\,!\,L(B)$.

3 The Descriptional Complexity of the Cut Operation

In this section we investigate the range of attainable complexities for the cut operation. In the first subsection we investigate the unary case and we show that some values may be unattainable. In the second subsection we study this problem for regular languages over an arbitrary alphabet, and we obtain a contiguous range of complexities from one up to the known upper bound already in the binary case.

3.1 The Cut Operation on Unary Regular Languages

When working with unary DFAs, we use the notational convention proposed by Nicaud in [20].

Every unary DFA consists of a tail path, which starts from the initial state, followed by a loop of one or more states. Let $A = (Q, \{a\}, \delta, q_0, F)$ be a unary DFA with $|Q| = n$. We can identify the states of A with integers from $[0, n-1]$ via $q \mapsto \min\{\, i \mid \delta(q_0, a^i) = q \,\}$. In particular the initial state q_0 is mapped to 0. Let $\ell = \delta(q_0, a^n)$. Then the unary DFA A with n states, loop number ℓ ($0 \leq \ell \leq n-1$), and set of final states F ($F \subseteq [0, n-1]$) is referred to as $A = (n, \ell, F)$. The following characterization of minimal unary DFAs is known.

Lemma 1 **([20]).** *A unary* DFA $A = (n, \ell, F)$ *is minimal if and only if*

1. *its loop is minimal, and*
2. *if $\ell \neq 0$, then states $n-1$ and $\ell - 1$ do not have the same finality, that is, exactly one of them is final.*

Now we are ready for our first result on the cut operation of unary regular languages represented by DFAs. In a series of lemmata we consider the state complexity α of the resulting language in increasing order of α. The first interval we are going to discuss is $[1, m]$.

Lemma 2. *Let $m, n \geq 1$ and $1 \leq \alpha \leq m$. There exist a minimal unary m-state* DFA *A and a minimal unary n-state* DFA *B such that the minimal* DFA *for $L(A)\,!\,L(B)$ has α states.*

Proof. The proof has five cases:

1. Let $m = 1$, so we must have $\alpha = 1$. Let A be the one-state DFA accepting the empty language and B be the minimal n-state DFA for $a^{n-1}a^*$. Then $L(A)\,!\,L(B) = \emptyset$ which is accepted by a minimal one-state DFA.
2. Let $m \geq 2$ and $n = 1$. Let A be the minimal m-state DFA for $a^{\alpha-1}(a^m)^*$ and B be the one-state DFA for a^*. The reachable part of the cut automaton $A\,!\,B$ consists of the tail of non-final states $(i, \bot)$ with $0 \leq i \leq \alpha - 2$ and the loop of final states $(i, 0)$ with $0 \leq i \leq m-1$. Since all the final states are equivalent, the minimal DFA for $L(A)\,!\,L(B)$ has α states.
3. Let $m, n \geq 2$ and $\alpha = 1$. Consider the unary languages $a^{m-1}a^*$ and $a^{n-1}a^*$ accepted by minimal DFAs A and B of m and n states, respectively. Then the reachable part of the cut automaton $A\,!\,B$ consists of the tail of non-final states $(i, \bot)$ with $1 \leq i \leq m-2$, and the loop consisting of a single non-final state $(m-1, 0)$; notice that 0 is a non-final state in B. Hence $L(A)\,!\,L(B)$ is the empty language accepted by a one-state DFA.
4. Let $m \geq 2$, $n = 2$, and $2 \leq \alpha \leq m$. Consider the unary languages K and L defined as follows. If $m - \alpha$ is even, then $K = \{\, a^{\alpha-2}, a^{m-2} \,\}$ and $L = a(aa)^*$, otherwise, $K = \{\, a^{\alpha-1}, a^{m-2} \,\}$ and $L = (aa)^*$. The minimal DFAs for K and L have m and 2 states, respectively. We have $K\,!\,L = a^{\alpha-1}(aa)^*$, which is accepted by a minimal α-state DFA.

5. Let $m \geq 2$, $n \geq 3$, and $2 \leq \alpha \leq m$. Consider the unary deterministic finite automata $A = (m, \alpha-2, [\alpha-1, m-1])$ and $B = (n, n-1, [0, n-2])$. By Lemma 1, the DFAs A and B are minimal. The reachable part of the cut automaton consists of the tail of $\alpha - 1$ non-final states and of the loop of $m - \alpha + 2$ final states. Hence the minimal DFA $(\alpha, \alpha-1, \{\alpha-1\})$ for $L(A) \,!\, L(B)$ has α states. $\square$

Our next interval is $[m+1, 2m-1]$; cf. $f_1(m,n)$ defined by (2) on page 4.

Lemma 3. *Let $m, n \geq 2$ and $m+1 \leq \alpha \leq 2m-1$. There exist a minimal unary m-state* DFA *A and a minimal unary n-state* DFA *B such that the minimal* DFA *for $L(A) \,!\, L(B)$ has α states.* $\square$

The last interval we are considering in this series of lemmata is $[n, m+n-2]$.

Lemma 4. *Let $m, n \geq 2$, $\alpha \geq m$, and $n \leq \alpha \leq m+n-2$. There exist a minimal unary m-state* DFA *A and a minimal unary n-state* DFA *B such that the minimal* DFA *for $L(A) \,!\, L(B)$ has α states.* $\square$

For certain values of m and n the intervals stated in the previous lemmata may not be contiguous. For instance, if we choose $m = 2$ and $n = 5$, then the intervals from Lemmata 2, 3, and 4 cover $\{1, 2, 3, 5\}$. Hence the value 4, which comes from the interval $[2m, n-1]$, is missing. In fact, we show that whenever this interval is nonempty, these values cannot be obtained by an application of the cut operation on minimal DFAs with an appropriate number of states.

Lemma 5. *Let $m, n \geq 2$ be numbers satisfying $2m \leq n-1$. Then for every α with $2m \leq \alpha \leq n-1$, there exist* no *minimal unary m-state* DFA *A and minimal unary n-state* DFA *B such that the minimal* DFA *for $L(A) \,!\, L(B)$ has α states.*

Proof. We discuss two cases depending on whether $L(A)$ is infinite or finite.

If $L(A)$ is infinite, then A must have a final state in its loop. Denote the size of loop in A by ℓ and the smallest final state in the loop of A by j. Consider the cut automaton $A \,!\, B$. Notice that its initial state is sent to the state $(j, 0)$ by the word a^j. Next, the state $(j, 0)$ is sent to itself by the word a^ℓ. It follows that $A \,!\, B$ is equivalent to a DFA $(j+\ell, j, F)$ for some set $F \subseteq [0, j+l-1]$. Since $j \leq m-1$ and $\ell \leq m$, the DFA for $L(A) \,!\, L(B)$ has at most $2m-1$ states.

If $L(A)$ is finite, then A has a loop in the non-final state $m-1$ and the state $m-2$ is final. Let $A = (m, m-1, F)$ and $B = (n, k, F')$ be minimal unary DFAs for some sets $F \subseteq [0, m-1]$ and $F' \subseteq [0, n-1]$. It follows that in the cut automaton $A \,!\, B$, the state $(m-2, 0)$ and the states $(m-1, j)$ with $1 \leq j \leq n-1$ are reachable. Two distinct states $(m-1, j)$ and $(m-1, j')$ are distinguishable by the same word as the states j and j' in B, and the state $(m-2, 0)$ and a state $(m-1, j)$ are distinguishable by the same word as 0 and j are distinguishable in B. It follows that the cut automaton has at least n reachable and pairwise distinguishable states, and the theorem follows. $\square$

Now let us summarize the results of this subsection; recall that the state complexity of the cut operation on unary languages is given by the function $f_1(m,n)$ defined by (2) on page 4 such that $f_1(1,n)=1$, $f_1(m,1)=m$, $f_1(m,n)=2m-1$ if $m,n \geq 2$ and $m \geq n$, and $f_1(m,n)=m+n-2$ if $m,n \geq 2$ and $m < n$.

Theorem 6 (Unary Case). *For every $m,n,\alpha \geq 1$ such that*

(i) $\alpha = 1$ *if* $m = 1$,
(ii) $1 \leq \alpha \leq m$ *if* $m \geq 2$ *and* $n = 1$, *or*
(iii) $1 \leq \alpha \leq 2m-1$ *or* $n \leq \alpha \leq m+n-2$ *if* $m,n \geq 2$,

there exist a minimal unary m-state DFA A *and a minimal unary n-state* DFA B *such that the minimal* DFA *for $L(A)\,!\,L(B)$ has α states. In the case of $m,n \geq 2$ and $2m \leq \alpha \leq n-1$, there do not exist minimal unary m-state and n-state* DFA*s A and B such that the minimal* DFA *for $L(A)\,!\,L(B)$ has α states.* □

3.2 The Cut Operation on Binary Regular Languages

Next we consider the range of state complexities of languages resulting from the cut operation on regular languages over an arbitrary alphabet. The aim of this subsection is to show that the entire range of complexities up to the known upper bound can be produced in this case, even for languages over a binary alphabet. First, we show that the numbers in $[1, m+n-2]$ are attainable in the binary case. The values in $[1, 2m-1]$ as well as the cases of $m=1$ or $n=1$ are covered by Theorem 6 since duplicating the symbols does not change the state complexity.

Lemma 7. *Let $m,n \geq 2$ and $2m \leq \alpha \leq m+n-2$. There exist a minimal binary m-state* DFA A *and a minimal binary n-state* DFA B *such that the minimal* DFA *for $L(A)\,!\,L(B)$ has α states.*

Proof. Notice that in this case we must have and $m < n$. Consider the binary DFA $A = ([0,m-1], \{a,b\}, \delta_A, 0, \{m-1\})$, where

$$\delta_A(i,a) = (i+1) \bmod m \quad \text{and} \quad \delta_A(i,b) = \begin{cases} (i+1) \bmod m, & \text{if } i \neq m-2, \\ m-2, & \text{otherwise.} \end{cases}$$

Next, consider the binary DFA $B = ([0,n-1], \{a,b\}, \delta_B, 0, \{m-1\})$, where

$$\delta_B(j,a) = (j+1) \bmod n \quad \text{and} \quad \delta_B(j,b) = \begin{cases} (j+1) \bmod n, & \text{if } j \neq \alpha - m, \\ m-1, & \text{otherwise.} \end{cases}$$

Both automata A and B are depicted in Fig. 1.

In the cut automaton $A\,!\,B$ we consider the following sets of states:

$$\mathcal{R}_1 = \{\, (i,\bot) \mid 0 \leq i \leq m-2 \,\} \cup \{(m-1,0)\} \cup \{\, (i,i+1) \mid 0 \leq i \leq m-3 \,\},$$
$$\mathcal{R}_2 = \{\, (m-2,j) \mid m-1 \leq j \leq \alpha - m \,\}.$$

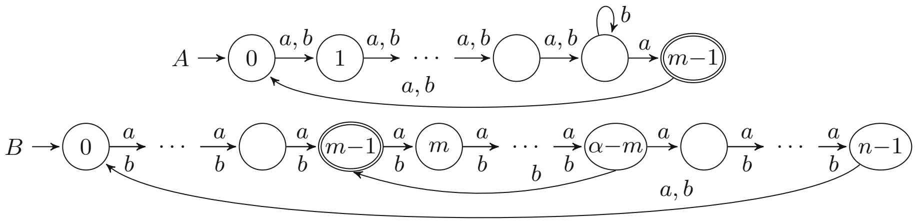

Fig. 1. The DFAs A (top) and B (bottom) for the case $m < n$ and $2m \leq \alpha \leq m+n-2$

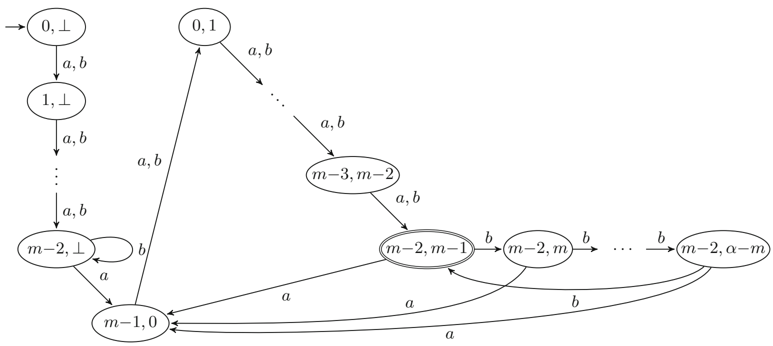

Fig. 2. The cut automaton for the DFAs in Fig. 1

Each state in $\mathcal{R}_1 \cup \{(m-2, m-1)\}$ is reached from $(0, \perp)$ by a word in a^*, and each state in $\mathcal{R}_2$ is reached from $(m-2, m-1)$ by a word in b^*. Figure 2 shows that no other state is reachable in the cut automaton.

To prove distinguishability, notice that two distinct states in $\mathcal{R}_1$ are distinguishable by a word in a^* and two distinct states in $\mathcal{R}_2$ are distinguishable by a word in b^*. The states $(i, \perp)$ in $\mathcal{R}_1$ are distinguishable from each state in $\mathcal{R}_2$ by a word in b^*. Every other state in $\mathcal{R}_1$ is distinguishable from each state in the set $\mathcal{R}_2$ by a word in a^*. Since $|\mathcal{R}_1 \cup \mathcal{R}_2| = \alpha$, our proof is complete. $\square$

Since the state complexity of the cut operation for regular languages in general is higher than those for unary languages, we have to consider the remaining interval $[m+n-1, (m-1)n+m]$. This is done in the following steps (cf. [8]):

1. First we show that some special values of α, corresponding to the number of states of the cut automaton in the first r rows and the first s columns, see Fig. 3, are attainable, namely $\alpha = 1 + (r-1)n + (m-r)s$ for some r, s with $2 \leq r \leq m$ and $1 \leq s \leq n$.
2. Then we show that all the remaining values of α in $[m+n-1, (m-1)n+1]$ are attainable.
3. Finally, we show that all the values of α in $[(m-1)n+2, (m-1)n+m]$ are attainable.

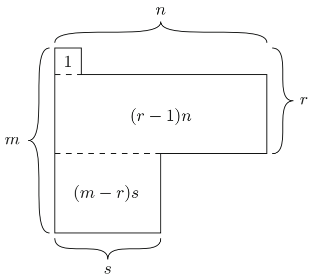

Fig. 3. A schematic drawing of the reachable part of the cut automaton

Let us start with the first task.

Lemma 8. *Let $m, n \geq 2$ and let r, s be any integers such that $2 \leq r \leq m$ and $1 \leq s \leq n$. Then there exist a minimal binary m-state* DFA $A_{r,s}$ *and a minimal binary n-state* DFA $B_{r,s}$ *such that the minimal* DFA *for $L(A_{r,s})\,!\,L(B_{r,s})$ has exactly $1 + (r-1)n + (m-r)s$ states.*

Proof. Our aim is to define the DFAs $A_{r,s} = ([0, m-1], \{a, b\}, \delta_A, 0, \{0\})$ and $B_{r,s} = ([0, n-1], \{a, b\}, \delta_B, 0, \{n-1\})$ in such a way that in the DFA $A_{r,s}\,!\,B_{r,s}$ the states in the following set would be reachable and pairwise distinguishable:

$$\begin{aligned} \mathcal{R} = \{(0,0)\} \cup \{\, (i,j) \mid 1 \leq i \leq r-1 \text{ and } 0 \leq j \leq n-1 \,\} \\ \cup \{\, (i,j) \mid r \leq i \leq m-1 \text{ and } 0 \leq j \leq s-1 \}. \end{aligned}$$

Moreover, we have to assure that no other state of the cut automaton is reachable. Because $|\mathcal{R}| = 1 + (r-1)n + (m-r)s$, the DFAs $A_{r,s}$ and $B_{r,s}$ will be the desired DFAs. To this aim, we define δ_A and δ_B as follows:

$$\delta_A(i, a) = (i+1) \bmod m \quad \text{and} \quad \delta_A(i, b) = \begin{cases} i, & \text{if } i \leq r-1; \\ r, & \text{if } i \geq r; \end{cases}$$

and

$$\delta_B(j, b) = (j+1) \bmod n \quad \text{and} \quad \delta_B(j, a) = \begin{cases} j, & \text{if } j \leq s-1; \\ s-1, & \text{if } j \geq s. \end{cases}$$

In the cut automaton $A_{r,s}\,!\,B_{r,s}$, the state $(0,0)$ is the initial state, and each state (i,j) in $\mathcal{R}$ is reached from $(0,0)$ by $a^i b^j$. To show that no other state is reachable, notice that each state (i,j) in $\mathcal{R}$ goes on a to a state (i', j') where $j' \leq s-1$, and it goes on b to a state (i'', j'') where $i'' \leq r-1$. Since both resulting states are in $\mathcal{R}$, no other state is reachable in the cut automaton.

It remains to prove the distinguishability of states in $\mathcal{R}$. The state $(0,0)$ and any other state in $\mathcal{R}$ are distinguishable by a word in b^*. Two states in different columns are distinguishable by a word in b^* since exactly one of them can be moved to the last column containing the final states of the cut automaton. Two states in different rows are distinguishable by a word in a^* since exactly one of them can be moved to the state $(0, 0)$. This proves distinguishability and concludes the proof. □

In the above lemma we obtained the values $\alpha_{r,s} = 1 + (r-1)n + (m-r)s$ in $[m+n-1, (m-1)n+1]$. We still need to get the values between $\alpha_{r,s}$ and $\alpha_{r+1,s}$ resp. $\alpha_{r,s+1}$. We have $\alpha_{r+1,s} - \alpha_{r,s} = n - s$ and $\alpha_{r,s+1} - \alpha_{r,s} = m - r$, so we need to obtain the complexities $\alpha_{r,s} + t$, where $1 \leq t \leq \min\{n-s, m-r\} - 1$. The next lemma produces these complexities.

Lemma 9. *Let $m, n \geq 2$ and let r, s be any integers such that $2 \leq r \leq m$ and $1 \leq s \leq n$. Moreover let t satisfy $1 \leq t \leq \min\{n-s, m-r\} - 1$. Then there exist a minimal binary m-state* DFA $A_{r,s,t}$ *and a minimal binary n-state* DFA $B_{r,s,t}$ *such that the minimal* DFA *for the language $L(A_{r,s,t}) \,!\, L(B_{r,s,t})$ has exactly $1 + (r-1)n + (m-r)s + t$ states.*

Proof. Let $\alpha_{r,s} = 1 + (r-1)n + (m-r)s$. Then in the cut automaton $A_{r,s} \,!\, B_{r,s}$ described in the previous proof, exactly $\alpha_{r,s}$ states are reachable and distinguishable. Our aim is to modify both automata in such a way that the resulting cut automaton has t more reachable states. To achieve this goal, we modify DFAs $A_{r,s}$ and $B_{r,s}$ as follows.

In $A_{r,s}$ we replace each transition $(r+i, b, r-1)$ by $(r+i, b, r+i-1)$, if $2 \leq i \leq t$ and i is even. Since $i \leq t \leq (m-r)-1$, we have $r+i \leq m-1$. In $B_{r,s}$ we replace each transition $(s+i, a, s-1)$ by $(s+i, a, s+i-1)$, if $1 \leq i \leq t$ and i is odd. Since $i \leq t \leq (n-s)-1$, we have $s+i \leq n-1$. Denote the resulting DFAs by $A_{r,s,t}$ and $B_{r,s,t}$, respectively. Consider the cut automaton $A_{r,s,t} \,!\, B_{r,s,t}$. Let $\mathcal{R}$ be the same set as in the previous proof. Then each state (i, j) in $\mathcal{R}$ is reachable from $(0, 0)$ by $a^i b^j$. Next, if i is odd, then each state $q_i = (r, s+i-1)$ is reached from $(r-1, s+i)$ by a, and otherwise, each state $q_i = (r+i-1, s)$ is reached from $(r+i, s-1)$ by b.

Now, let us show that no other state is reachable. Notice that each state in $\mathcal{R}$ goes either to a state in $\mathcal{R}$ or to a state in $\{q_1, q_2, \ldots, q_t\}$ on a and b; each state $(r-1, s+i)$ with i even goes to $(r, s-1)$ on a, and each state $(r+i, s-1)$ with $i = 0$ or i odd goes to $(r-1, s)$ on b. Next, each state q_i with i odd goes to the state $(r+1, s-1)$ on a and to a state in row $r-1$ on b. Finally, each state q_i with i even goes to a state in column $s-1$ on a and to the state $(r-1, s+1)$ on b. Since all the resulting states are in $\mathcal{R} \cup \{q_1, q_2, \ldots, q_t\}$, no other state is reachable in the cut automaton.

The proof of distinguishability is exactly the same as in Lemma 8. □

In the two lemmata above, we have produced all the complexities in the range from $m+n-1$ to $(m-1)n+1$. It remains to show that the complexities in $[(m-1)n+2, (m-1)n+m]$ are attainable.

Lemma 10. *Let $m, n \geq 2$ and $(m-1)n+2 \leq \alpha \leq (m-1)n+m$. There exist a minimal binary m-state* DFA A *and a minimal binary n-state* DFA B *such that the minimal* DFA *for $L(A) \,!\, L(B)$ has exactly α states.*

Proof. We have $\alpha = (m-1)n + 1 + \beta$ for some β with $1 \leq \beta \leq m-1$. Let A be a minimal m-state DFA over $\{a, b\}$ that accepts the words in which the number of a's modulo m is β. Let B be a minimal n-state DFA over $\{a, b\}$ that accepts the words in which the number of b's modulo n is $n-1$.

Consider the cut automaton $A \,!\, B$. Denote

$$\mathcal{R}_1 = \{\, (i, \perp) \mid i \in [0, \beta - 1] \,\} \cup \{(\beta, 0)\},$$

and

$$\mathcal{R}_2 = \{\, (i, j) \mid i \in [0, \beta - 1] \cup [\beta + 1, m - 1] \text{ and } j \in [0, n - 1] \,\}.$$

Notice that each state $(i, \perp)$ in $\mathcal{R}_1$ is reachable from the initial state $(0, \perp)$ by a^i, and each state $(i, 0)$ is reachable by a^{m+i}. Each state (i, j) in $\mathcal{R}_2$ is reached from $(0, 0)$ by $a^i b^j$. Since the state β is a final state in A, it follows from the construction of the cut automaton that no state $(i, \perp)$ with $i \geq \beta$ and no state in row β except for $(\beta, 0)$ is reachable.

To prove distinguishability, let p and q be two different states in $\mathcal{R}_1 \cup \mathcal{R}_2$. If $p \in \mathcal{R}_1$ and $q \in \mathcal{R}_2$, then p is a non-final state with a loop on b, while a word in b^* is accepted from q. If both p and q are in $\mathcal{R}_1$, then a word in a^* leads one of them to the state $((\beta + 1) \bmod m, 0)$ in $\mathcal{R}_2$, while it leads the second one to a state in $\mathcal{R}_1$, and the resulting states are distinguishable as shown above. Finally, let p and q be two states in $\mathcal{R}_2$. If they are in different columns, then a word in b^* distinguishes them. If p and q are in different rows, then a word in a^* leads one of them to the state $(\beta, 0)$ in $\mathcal{R}_1$, and it leads the second one to a state in $\mathcal{R}_2$. $\square$

The next theorem summarizes the results of this section; recall that the state complexity of the cut operation is given by the function $f(m, n)$ defined by (1) on page 4 such that $f(m, 1) = m$ and $f(m, n) = (m-1)n + m$ if $n \geq 2$.

Theorem 11 (General Case). *Let $m, n \geq 1$ and $f(m, n)$ be the state complexity of the cut operation. For each α such that $1 \leq \alpha \leq f(m, n)$, there exist a minimal binary m-state* DFA *A and a minimal binary n-state* DFA *B such that the minimal* DFA *for $L(A) \,!\, L(B)$ has α states.* $\square$

Observe that this theorem solves the magic number problem for the cut operation for every alphabets of size at least two by duplicating input symbols.

4 Conclusions

We examined the state complexity of languages resulting from the cut operation on minimal DFAs with m and n states. We showed that the range of state complexities of languages resulting from the cut operation is contiguous from one up to the known upper bound for every alphabet of size at least two. Our results in the unary case are different. We proved that no value from $2m$ up to $n-1$ is attainable by the state complexity of the cut of two unary languages represented by minimal deterministic finite automata with m and n states. All the remaining values up to the known upper bound are attainable. This means that the problem of finding all attainable complexities for the cut operation is completely solved for every size of alphabet. To the best of our knowledge, the cut operation is the first operation where this is the case.

Acknowledgments. We thank Juraj Šebej and Jozef Jirásek Jr. for their help on border values in our theorems. Moreover, also thanks to Galina Jirásková for her support and to all who helped us to improve the presentation of the paper.

References

1. Berglund, M., Björklund, H., Drewes, F., van der Merwe, B., Watson, B.: Cuts in regular expressions. In: Béal, M.-P., Carton, O. (eds.) DLT 2013. LNCS, vol. 7907, pp. 70–81. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-38771-5_8
2. Čevorová, K.: Kleene star on unary regular languages. In: Jurgensen, H., Reis, R. (eds.) DCFS 2013. LNCS, vol. 8031, pp. 277–288. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-39310-5_26
3. Čevorová, K., Jirásková, G., Krajňáková, I.: On the square of regular languages. In: Holzer, M., Kutrib, M. (eds.) CIAA 2014. LNCS, vol. 8587, pp. 136–147. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-08846-4_10
4. Drewes, F., Holzer, M., Jakobi, S., van der Merwe, B.: Tight bounds for cut-operations on deterministic finite automata. Fundam. Inform. **155**(1–2), 89–110 (2017)
5. Geffert, V.: Magic numbers in the state hierarchy of finite automata. Inf. Comput. **205**(11), 1652–1670 (2007)
6. Geffert, V.: State hierarchy for one-way finite automata. J. Autom. Lang. Comb. **12**(1–2), 139–145 (2007)
7. Harrison, M.A.: Introduction to Formal Language Theory. Addison-Wesley, Boston (1978)
8. Hricko, M., Jirásková, G., Szabari, A.: Union and intersection of regular languages and descriptional complexity. In: DCFS 2005, pp. 170–181. Università degli Studi di Milano, Italy (2005)
9. Iwama, K., Kambayashi, Y., Takaki, K.: Tight bounds on the number of states of DFAs that are equivalent to n-state NFAs. Theoret. Comput. Sci. **237**(1–2), 485–494 (2000)
10. Iwama, K., Matsuura, A., Paterson, M.: A family of NFA's which need $2^n - \alpha$ deterministic states. In: Nielsen, M., Rovan, B. (eds.) MFCS 2000. LNCS, vol. 1893, pp. 436–445. Springer, Heidelberg (2000). https://doi.org/10.1007/3-540-44612-5_39
11. Jirásek, J., Jirásková, G., Szabari, A.: Deterministic blow-ups of minimal nondeterministic finite automata over a fixed alphabet. Internat. J. Found. Comput. Sci. **19**(3), 617–631 (2008)
12. Jirásková, G.: Deterministic blow-ups of minimal NFA's. RAIRO-ITA **40**(3), 485–499 (2006)
13. Jirásková, G.: Concatenation of regular languages and descriptional complexity. Theory Comput. Syst. **49**(2), 306–318 (2011)

14. Jirásková, G.: Magic numbers and ternary alphabet. Int. J. Found. Comput. Sci. **22**(2), 331–344 (2011)
15. Jirásková, G., Palmovský, M., Šebej, J.: Kleene closure on regular and prefix-free languages. In: Holzer, M., Kutrib, M. (eds.) CIAA 2014. LNCS, vol. 8587, pp. 226–237. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-08846-4_17
16. Jirásková, G., Szabari, A., Šebej, J.: The complexity of languages resulting from the concatenation operation. In: Câmpeanu, C., Manea, F., Shallit, J. (eds.) DCFS 2016. LNCS, vol. 9777, pp. 153–167. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-41114-9_12
17. Lupanov, O.B.: A comparison of two types of finite automata. Problemy Kibernetiki **9**, 321–326 (1963). (in Russian). German translation: Über den Vergleich zweier Typen endlicher Quellen. Probleme der Kybernetik 6, 328–335 (1966)
18. Meyer, A.R., Fischer, M.J.: Economy of description by automata, grammars, and formal systems. In: Proceedings of the 12th Annual Symposium on Switching and Automata Theory, pp. 188–191. IEEE Computer Society Press (1971)
19. Moore, F.R.: On the bounds for state-set size in the proofs of equivalence between deterministic, nondeterministic, and two-way finite automata. IEEE Trans. Comput. **C–20**, 1211–1219 (1971)
20. 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). https://doi.org/10.1007/3-540-48340-3_21
21. Rabin, M.O., Scott, D.: Finite automata and their decision problems. IBM J. Res. Dev. **3**, 114–125 (1959)
22. Šebej, J.: Reversal on regular languages and descriptional complexity. In: Jurgensen, H., Reis, R. (eds.) DCFS 2013. LNCS, vol. 8031, pp. 265–276. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-39310-5_25
23. Yershov, Y.L.: On a conjecture of V. A. Uspenskii. Algebra i logika **1**, 45–48 (1962). (in Russian)
24. Yu, S., Zhuang, Q., Salomaa, K.: The state complexities of some basic operations on regular languages. Theoret. Comput. Sci. **125**(2), 315–328 (1994)