

ON THE ACCEPTING STATE COMPLEXITY OF OPERATIONS ON PERMUTATION AUTOMATA

CHRISTIAN RAUCH* AND MARKUS HOLZER

Abstract. We investigate the accepting state complexity of deterministic finite automata for regular languages obtained by applying one of the following operations on languages accepted by permutation automata: union, quotient, complement, difference, intersection, Kleene star, Kleene plus, and reversal. The paper thus joins the study of the accepting state complexity of regularity preserving language operations which was initiated in [J. Dassow, *J. Autom., Lang. Comb.* **21** (2016) 55–67]. We show that for almost all of the above-mentioned operations, except for reversal and quotient, there is no difference in the accepting state complexity for permutation automata compared to deterministic finite automata in general. For both reversal and quotient we prove that certain accepting state complexities cannot be obtained; these numbers are called “magic” in the literature. Moreover, we solve the left open accepting state complexity problem for the intersection of unary languages accepted by permutation automata and deterministic finite automata in general.

Mathematics Subject Classification. 68Q04, 68Q10, 68Q45.

Received January 30, 2023. Accepted September 19, 2023.

1. INTRODUCTION

The state complexity of a regular language is a classical well-understood descriptonal complexity measure of finite state systems, that is defined to be the number of states of the smallest, either deterministic or nondeterministic, finite automaton that recognizes it. It has been studied from different perspectives in the literature like, for instance, (i) for regular languages in general and for certain sub-families, (ii) for converting nondeterministic finite automata to equivalent deterministic finite automata, and (iii) for operations, called the operational complexity, on regular languages in general and sub-families thereof. For a brief survey on the subject we refer to, *e.g.*, [4].

Recently, the accepting state complexity of a regular language was introduced in [3]. It is defined to be the minimal number of accepting states needed for a finite state device, either deterministic or nondeterministic, that accepts it. While the accepting state complexity forms a strict hierarchy of languages classes for deterministic finite automata, it collapses for nondeterministic state devices, since every regular language not containing the empty word is accepted by a nondeterministic finite automaton with a single final state. If the empty word belongs to the language, the nondeterministic accepting state complexity is at most two. Thus, the conversion

Keywords and phrases: Permutation automata, language operations, accepting state complexity, descriptonal complexity.

Institut für Informatik, Universität Giessen, Arndtstr. 2, 35392 Giessen, Germany.

* Corresponding author: christian.rauch@informatik.uni-giessen.de

from nondeterministic to equivalent deterministic finite automata can produce unbounded deterministic accepting state complexity for a regular language. Moreover, the operational accepting state complexity was studied in [7]. The obtained results on the accepting state complexity prove that this measure is significantly different to the original state complexity. What is missing for the accepting state complexity is a study for certain sub-families of the regular languages in order to better understand the intrinsic behaviour of this measure.

We close this gap by studying the operational accepting state complexity for the class of permutation automata (PFAs), which accept the so called p-regular languages, also named pure-group languages. This language family is of particular interest from an algebraic point of view since their syntactic monoid induces a group. Additionally permutation automata together with permutation-reset automata play a key role in the decomposition of deterministic finite automata (DFAs), see, *e.g.*, [11]. It is also worth to mention that the class of p-regular languages was one of the first subclasses of the regular languages for which the star height problem was shown to be decidable, see, *e.g.*, [2]. Recently, the family of p-regular languages, and thus PFAs, gained renewed interest. For instance, in [10] the decomposing of PFAs into the intersection of smaller automata of the same kind was investigated. Moreover the operational state complexity on PFAs was studied in [8]. Up to our knowledge the operational accepting state complexity of p-regular languages was not investigated so far. We study this problem by examining the following question:

Let three non-negative integers m , n , and α and a regularity preserving language operation \circ be given. Are there minimal permutation automata A and B with accepting state complexity m and n , respectively, such that the language $L(A) \circ L(B)$ is accepted by a minimal deterministic finite automaton with α accepting states?

Following the terminology of [9] we call values α “magic” if there are no such automata A and B . The following results were shown in [3] and [7] for the operational accepting state complexity on languages accepted by DFAs—for accepting state complexities m and n one can obtain all values from the given number set for α :

- Complement: $\mathbb{N} \cup \{0 \mid m = 1\}$.
- Kleene star and Kleene plus: \mathbb{N} .
- Union: \mathbb{N} .
- Set difference: \mathbb{N} .
- Intersection: $[0, mn]$ (if the input alphabet is at least binary).
- Reversal: \mathbb{N} .
- Quotient: $\mathbb{N} \cup \{0\}$.

One may have noticed that for none of the above mentioned operations magic numbers exist.¹ It is worth mentioning that these results were obtained without the use of PFA witnesses.

We generalize these results for the operations complement, union, set difference, Kleene star, and Kleene plus to the family of languages accepted by PFAs. This means even though PFAs are restricted in their expressive power compared to arbitrary DFAs, there are no magic numbers for the accepting state complexity of PFAs w.r.t. the above mentioned operations. When considering the reversal operation a significant difference appears. While for DFAs the reversal operation induces the whole set \mathbb{N} as accepting state complexities as mentioned above, this is not the case for PFAs, where we can prove that the number $\alpha = 1$ is magic for every $m \geq 2$. In fact, we prove that for $m = 2$ no other magic number as $\alpha = 1$ exists. Whether this is also true for larger m is left open. Yet another difference in the accepting state complexity comes from the quotient operation. Here it turns out that for unary languages accepted by PFAs only the range $[1, mn]$ is obtainable for the accepting state complexity. This is entirely different compared to the general case. Finally, the unary case for the accepting state complexity of the intersection operation for DFAs in general was left open in [7]. We close this gap by considering this problem in detail. In this way, we identify a whole range of magic numbers for the intersection

¹The intersection of two languages L_1 and L_2 of accepting state complexity m and n is accepted by the cross product of the minimal DFAs accepting L_1 and L_2 . So the accepting state complexity is directly bounded by mn . Therefore the numbers greater mn are not of interest.

of unary languages accepted by PFAs and extend this result to the case of DFAs, solving the left open problem mentioned above.

2. PRELIMINARIES

Let \mathbb{N} denote the set of all positive integers and $\mathbb{N}_{\geq x}$ ($\mathbb{N}_{\leq x}$, respectively) the set of all positive integers that are greater or equal x (less or equal x , respectively).

We recall some definitions on finite automata as contained in [5]. Let Σ^* denote the set of all words over the finite alphabet Σ . The *empty word* is denoted by ε . Further, we denote the set $\{i, i+1, \dots, j\}$ by $[i, j]$, if i and j are integers. A *deterministic finite automaton* (DFA) is a quintuple $A = (Q, \Sigma, \cdot, q_0, F)$, where Q is the finite set of *states*, Σ is the finite set of *input symbols*, $q_0 \in Q$ is the *initial state*, $F \subseteq Q$ is the set of *accepting states*, and the *transition function* \cdot maps $Q \times \Sigma$ to Q . The *language accepted* by the DFA A is defined as

$$L(A) = \{w \in \Sigma^* \mid q_0 \cdot w \in F\},$$

where the transition function is recursively extended to a map $Q \times \Sigma^*$ onto Q . Obviously, every letter $a \in \Sigma$ induces a mapping on the state set Q to Q by $q \mapsto q \cdot a$, for every $q \in Q$. A DFA is *unary*, if the input alphabet Σ is a singleton set, that is, $\Sigma = \{a\}$, for some input symbol a . Moreover, if every letter of the automaton induces only permutations on the state set, then we simply speak of a *permutation automaton* (PFA).

As usual we denote the *state complexity* of a language L accepted by a DFA by

$$\text{sc}(L) = \min\{\text{sc}(A) \mid A \text{ is a DFA with } L = L(A)\},$$

where $\text{sc}(A)$ refers to the number of states of the automaton A . Similarly we define the measure $\text{asc}(L)$ as the *accepting state complexity* of a language L accepted by a DFA, where $\text{asc}(A)$ refers to the number of final states of the automaton A .

An automaton is *minimal* (a-minimal, respectively) if it admits no smaller equivalent automaton w.r.t. the number of states (final states, respectively). For DFAs both properties can be easily verified. Minimality can be shown if all states are reachable from the initial state and all states are pairwise inequivalent. For a-minimality the following result shown in [3] applies:

Theorem 2.1. *Let L be a language accepted by a minimal DFA A . Then $\text{asc}(L) = \text{asc}(A)$.*

In order to characterize the behaviour of complexities under operations we introduce the following notation: for $c \in \{\text{sc}, \text{asc}\}$, a k -ary regularity preserving operation \circ on languages, and natural numbers n_1, n_2, \dots, n_k , we define

$$g_{\circ}^c(n_1, n_2, \dots, n_k)$$

as the set of all integers r such that there are k regular languages L_1, L_2, \dots, L_k with $c(L_i) = n_i$, for $1 \leq i \leq k$, and $c(\circ(L_1, L_2, \dots, L_k)) = r$. In case we only consider unary languages L_1, L_2, \dots, L_k we simply write $g_{\circ}^{c,u}$ instead. When restricting the underlying languages to, e.g., be accepted by permutation automata (PFAs), we indicate this by writing $g_{\circ, \text{PFA}}^c$ and $g_{\circ, \text{PFA}}^{c,u}$, respectively.

In order to explain the notation we give a small example.

Example 2.2. Consider the unary operation C of complementation of languages. It is obvious that

$$g_C^{\text{sc}}(m) = \{m\}, \quad \text{for } m \geq 1.$$

On the other hand, when we consider the accepting state complexity, in [3], Theorem 6 the following behaviour

$$g_C^{\text{asc}}(m) = \begin{cases} \{1\}, & \text{if } m = 0, \\ \{0\} \cup \mathbb{N}, & \text{if } m = 1, \\ \mathbb{N}, & \text{otherwise,} \end{cases}$$

for the complementation was proven. Moreover, it is easy to see that

$$g_C^{\text{sc},u}(m) = g_C^{\text{sc}}(m) \quad \text{and} \quad g_C^{\text{asc},u}(m) = g_C^{\text{asc}}(m)$$

holds. □

In the constructions to come, note that we will use the mod operation such that $x \bmod y + z$ is the same as $(x \bmod y) + z$ and not equal to $x \bmod (y + z)$, but $x + y \bmod z$ is the same as $(x + y) \bmod z$. We use \div for the integer division and the symbol $/$ for the ordinary division.

3. RESULTS

We investigate the accepting state complexity of various regularity preserving language operations such as union, quotient, complement, difference, intersection, Kleene star, Kleene plus, and reversal on languages accepted by permutation automata. Before we start our investigation we introduce a useful notion for *unary* permutation automata by strings. Since a unary permutation automaton consists of a cycle only, it suffices to encode the finality of these states by a binary string. This is done as follows: a word $w \in \{0, 1\}^+$ with $w = a_0 a_1 \dots a_{|w|-1}$, for $a_i \in \{0, 1\}$ and $0 \leq i \leq |w| - 1$, describes the permutation automaton

$$A_w = (\{0, 1, \dots, |w| - 1\}, \{a\}, \cdot, 0, \{i \mid 0 \leq i \leq |w| - 1 \text{ and } a_i = 1\})$$

with

$$i \cdot a = \begin{cases} i + 1, & \text{for } 0 \leq i < |w| - 1, \\ 0, & \text{otherwise.} \end{cases}$$

It is clear that there is a bijection between words in $\{0, 1\}^+$ with all unary PFAs. Thus, we can identify words with PFAs and *vice versa*. Now we are ready for the investigation of the accepting state complexity of certain operations on PFAs.

3.1. Complementation

The complement of a language accepted by a finite automaton can be obtained by simply interchanging accepting and non-accepting states. Hence, the state complexity of a language accepted by a finite automaton, regardless whether the automaton is a permutation automaton or not, is the same. The results on the *accepting state complexity* for unrestricted DFAs was presented in Example 2.2. Next we show that this result even holds for PFAs.

Theorem 3.1. *We have $g_{C,\text{PFA}}^{\text{asc},u}(m) = g_{C,\text{PFA}}^{\text{asc}}(m) = g_C^{\text{asc}}(m) = g_C^{\text{asc},u}(m)$.*

Proof. It suffices to show

$$\left. \begin{array}{ll} \{1\}, & \text{if } m = 0, \\ \{0\} \cup \mathbb{N}, & \text{if } m = 1, \\ \mathbb{N}, & \text{otherwise,} \end{array} \right\} \subseteq g_{C,\text{PFA}}^{\text{asc},u}(m),$$

due to the obvious inclusions

1. $g_{C,\text{PFA}}^{\text{asc},u}(m) \subseteq g_{C,\text{PFA}}^{\text{asc}}(m)$,
2. $g_{C,\text{PFA}}^{\text{asc},u}(m) \subseteq g_C^{\text{asc},u}(m)$,
3. $g_{C,\text{PFA}}^{\text{asc}}(m) \subseteq g_C^{\text{asc}}(m)$, and
4. the equality $g_C^{\text{asc},u}(m) = g_C^{\text{asc}}(m)$ from [3].

Let $L \subseteq a^*$ with $\text{asc}(L) = m$. If $m = 0$ the language $L = \emptyset$. Hence the complement of L is equal to a^* . These two languages can be accepted by single state PFAs, which either consists of a non-accepting or accepting state, respectively. Thus, we have $g_{C,\text{PFA}}^{\text{asc},u}(0) = \{1\}$. Next let $m \geq 1$. We consider the unary PFA A_w induced by the word $w = 1^m 0^\ell$, for $\ell \geq 0$. Clearly, A_w is minimal if and only if $\ell \geq 1$ or $\ell = 0$ and $m = 1$. Therefore, by the standard complementation construction that changes accepting to non-accepting states and *vice versa*, we conclude that $\{0\} \cup \mathbb{N} \subseteq g_{C,\text{PFA}}^{\text{asc},u}(1)$ and $\mathbb{N} \subseteq g_{C,\text{PFA}}^{\text{asc},u}(m)$, for $m \geq 2$. This proves the stated claim. \square

3.2. Kleene star and Kleene plus

Next we study the accepting state complexity of the Kleene star and the Kleene plus operations for permutation automata. We want to mention that the Kleene closure of a p-regular language cannot be accepted by a PFA in general, see for example [8]. First we prove a useful relation between PFAs and the languages which they accept.

Lemma 3.2. *Let I be a finite set of non-negative integers and j be a number which is greater than the biggest number in I . The language $\bigcup_{i \in I} a^i (a^j)^*$ can be accepted by the PFA*

$$A = (\{q_0, q_1, \dots, q_{j-1}\}, \{a\}, \cdot_A, q_0, \{q_i \mid i \in I\})$$

with $q_i \cdot_A a = q_{i+1 \bmod j}$. Additionally A is minimal if there is no divisor t of j such that for every $i \in I$ the number $(i+t) \bmod j$ is in I .

Proof. The tedious details for the first statement are left to the reader. We prove the second statement by contradiction. So assume that there is a divisor t of j such that for every $i \in I$ the number $(i+t) \bmod j$ is in I . Let A be minimal. Then for every pair of states there is a word w , which distinguishes them, *i.e.*, maps one of the states onto an accepting and the other one onto a non-accepting state. This includes the states q_i and $q_{i+t \bmod j}$. Since $w \in a^*$ we can assume $w = a^k$, for a non-negative integer k . Due to the definition of A the word w maps the states q_i and $q_{i+t \bmod j}$ onto $q_{i+k \bmod j}$ and $q_{i+t+k \bmod j}$, which are either both in I or are both not in I . This contradicts the assumption that w distinguishes q_i and $q_{i+t \bmod j}$. \square

We will use this result to prove that no magic numbers exist for the accepting state complexity of the Kleene star operation.

Theorem 3.3. *We have*

$$g_*^{\text{asc},u}(m) = g_{*,\text{PFA}}^{\text{asc},u}(m) = \begin{cases} \{1\}, & \text{if } m = 0, \\ \mathbb{N}, & \text{otherwise.} \end{cases}$$

Proof. For $m = 0$ we observe that $\emptyset^* = \{\epsilon\}$, for ϵ being the empty word. So the first statement follows. For the second one we distinguish whether α or m are equal to one. We distinguish four cases, where in each case we use Lemma 3.2 to show that the defined language has accepting state complexity m :

1. Case $\alpha = 1$ and $m > 1$: The language $L = \bigcup_{i=1}^m a^i (a^{m+1})^*$ has accepting state complexity m and its Kleene star is equal to Σ^* which has accepting state complexity one.
2. Case $\alpha = 1$ and $m = 1$: The language of the previous case can also be used if $m = 1$.

3. Case $\alpha > 1$ and $m > 1$: Let

$$L = a^2(a^{2(\alpha-1)+m+1})^* \cup \bigcup_{i=1}^{m-1} a^{2(\alpha-1)+i}(a^{2(\alpha-1)+m+1})^*.$$

Then L^* is equal to $\bigcup_{i=0}^{\alpha-2} a^{2i} \cup a^{2(\alpha-1)}\Sigma^*$. In turn $\bigcup_{i=0}^{\alpha-2} a^{2i} \cup a^{2(\alpha-1)}\Sigma^*$ can be accepted by a unary DFA which has a tail of length $2(\alpha - 1)$ and a cycle formed by one state, where all states on positions with an even number are accepting, if we start counting by zero.

4. Case $\alpha > 1$ and $m = 1$: Define $L = a^2(a^{2\alpha-3})^*$. Then the language L^* is equal to $\bigcup_{i=0}^{\alpha-2} a^{2i} \cup a^{2(\alpha-1)}\Sigma^*$, which is the Kleene star language from the previous case.

Thus in all cases the Kleene closure has accepting state complexity α which completes the proof. \square

By taking into account that for every language L the empty word ϵ is in L^+ if and only if $\epsilon \in L$, with a small adjustment of the used languages for the previous theorem we obtain the following corollary for the Kleene plus operation.

Corollary 3.4. *We have*

$$g_+^{\text{asc},u}(m) = g_{+,PFA}^{\text{asc},u}(m) = \begin{cases} \{0\}, & \text{if } m = 0, \\ \mathbb{N}, & \text{otherwise.} \end{cases}$$

Proof. Clearly, the cases where α equals one does not depend on the fact that $a^0 = \epsilon$ is added to the set of accepted words, *i.e.*, $L^+ = (\bigcup_{i=1}^m a^i(a^{m+1})^*)^+ = \Sigma^+$ which also has accepting state complexity one. For the cases $\alpha > 1$, $m > 1$ and $\alpha > 1$, $m = 1$ we make a little shift for the input languages, namely we set

$$L = a^2(a^{2\alpha+m})^* \cup \bigcup_{i=1}^{m-1} a^{2\alpha+i}(a^{2\alpha+m})^*$$

(and $L = a^2(a^{2\alpha-1})^*$, for $\alpha > 1$ and $m = 1$, respectively) which results in the language $\bigcup_{i=1}^{\alpha-1} a^{2i} \cup a^{2\alpha}\Sigma^*$. \square

3.3. Union

In this subsection we extend the results for the accepting state complexity from [3] for the union operation to the class of permutation automata. For DFAs in general the following result was shown in [3], Theorem 10:

$$g_{\cup}^{\text{asc},u}(m, n) = g_{\cup}^{\text{asc}}(m, n) = \begin{cases} \{m\}, & \text{if } n = 0, \\ \{n\}, & \text{if } m = 0, \\ \mathbb{N}, & \text{otherwise,} \end{cases}$$

and since the union operation is commutative $g_{\cup}^{\text{asc},u}(m, n) = g_{\cup}^{\text{asc},u}(n, m)$ and $g_{\cup}^{\text{asc}}(m, n) = g_{\cup}^{\text{asc}}(n, m)$. Note the languages that prove these results are *not* accepted by any PFA.

We will prove that except for the special cases $m = 0$ or $n = 0$ every accepting state complexity can be reached also for unary alphabets. Therefore the reachable numbers coincide in the cases that the input DFAs are restricted or not. We split this into three theorems which show that $\mathbb{N}_{\leq \min\{m,n\}}$, $\mathbb{N}_{\geq \max\{m,n\}}$, and the interval $[\min\{m, n\} + 1, \max\{m, n\} - 1]$ are reachable. We start with the upper range.

Theorem 3.5. For $m, n \geq 1$ we have $\mathbb{N}_{\geq \max\{m, n\}} \subseteq g_{\cup, PFA}^{asc, u}(m, n)$.

We split the proof of this theorem into two lemmata, which show the reachability of smaller intervals (Lem. 3.6) and that the union of those intervals equals the whole range $\mathbb{N}_{\geq \max\{m, n\}}$ (Lem. 3.7).

Lemma 3.6. Let $m \geq n \geq 1$, $i \geq 1$, and $\alpha \in [\max\{in, m\}, in + m]$. There are minimal unary PFAs A and B with accepting state complexity m and n , respectively, such that $L(A) \cup L(B)$ has accepting state complexity α .

Proof. For this proof we distinguish the cases whether $\alpha = m$, or the value of α lies in the interval $\alpha \in [\max\{in, m\} + 1, in + m]$, or $\alpha = in$.

1. Case $\alpha = m$: We simply choose $A = A_w$ and $B = A_{w'}$, for $w = 1^m 0$ and $w' = 1^n 0^{m+1-n}$. It is not hard to see that A and B are minimal and that $L(A) \cup L(B) = L(A)$.
2. Case $\alpha \in [\max\{in, m\} + 1, in + m]$: Let B be a minimal PFA which consists of a cycle of length k and has accepting state complexity n , for a positive integer $k > n$. Let A be a minimal PFA of size ki with m accepting states. If we pick the m positions of the accepting states of A such that A is minimal it is not hard to see that $L(A) \cup L(B)$ can be accepted by the minimal PFA C , which is obtained from A by adding some of its states to its set of accepting states. Those states are the ni states in

$$\widetilde{F}_B := \{q_{z+kj} \mid q_z \in F_B, j \in [0, i-1]\},$$

where F_B denotes the set of accepting states of B . One may say that B gets replicated and the union of the sets of accepting states of each of those replications forms \widetilde{F}_B .

Now Lemma 3.2 implies the minimality of A and C since we can choose k as any integer greater n and we can place one accepting state q in A and therefore in C such that the conditions of the lemma are fulfilled for A and C , which is possible if q is not in the above mentioned set of added accepting states.

3. Case $\alpha = in$: For preserving the minimality of A we alter the construction of the previous case a little bit. Namely we set the length of A to be $k(i-1)$, which causes \widetilde{F}_B to consist of $n(i-1)$ accepting states. Since $\max\{ni, m\} = ni \geq m$ we observe that $|\widetilde{F}_B| = n(i-1) = ni - n \geq m - n$. Because $m \geq n$ we can choose $m - n$ accepting states of A to be in \widetilde{F}_B ² and n states to be not in \widetilde{F}_B . Additionally we can also apply Lemma 3.2 here and obtain that C has $\alpha = ni = n(i-1) + n$ accepting states.

Lastly, we want to point out that for $i = 1$ there is no need to use the adapted construction from above because if we want to reach $1n$ the number $1n = n$ must be equal to m because $m \geq n$.

□

The next lemma shows that the union of the intervals which are reachable due to the previous lemma is again a consecutive interval.

Lemma 3.7. For $m \geq n$ holds $\bigcup_{i \in \mathbb{N}} [\max\{in, m\}, in + m] = \mathbb{N}_{\geq m}$.

Proof. If we look at the integer interval $[\max\{in, m\}, in + m]$ either $in < m$ or $in \geq m$ holds. In the first case $[\max\{in, m\}, in + m]$ is equal to $[m, in + m]$. This interval has a non-empty intersection with $[\max\{(i+1)n, m\}, (i+1)n + m]$, if $(i+1)n \leq m$. For $(i+1)n > m$ we observe that $[\max\{(i+1)n, m\}, (i+1)n + m]$ is equal to $[(i+1)n, (i+1)n + m]$ but this intersects non-emptily with $[m, in + m] \supseteq [m, in + n] = [m, (i+1)n]$. So in both cases $[\max\{(i+1)n, m\}, (i+1)n + m]$ and $[\max\{in, m\}, in + m]$ intersect non-trivially.

It remains to show that also for $in \geq m$ the ranges have a non-empty intersection. For $in \geq m$ we see that the intersection of the two ranges above is equal to

$$[in, in + m] \cap [(i+1)n, (i+1)n + m].$$

²This cannot cause a problem for the minimality of A since $m \geq n \geq 1$.

Since $[in, in + m] \supseteq [in, (i + 1)n]$ holds due to the assumption that $m \geq n$ we see that the intersection of those intervals must be non-empty, too. In summary we see that the intervals of the union intersect non-trivially for all $i \in \mathbb{N}$ and $i + 1$ so the claim of the lemma follows. \square

It is not hard to see that the Lemmata 3.6 and 3.7 are symmetric in n and m so together they prove Theorem 3.5. Next we show that the lower range is reachable, too.

Theorem 3.8. *For $m, n \geq 1$ we have $[1, \min\{m, n\}] \subseteq g_{\cup, PFA}^{asc, u}(m, n)$.*

Proof. Since the following constructions can be used symmetrically we will prove the statement only for $m \geq n$. Set $A = A_w$ and $B = A_{w'}$, for

$$w = (1^\alpha 0)^{m \div \alpha} 1^{m \bmod \alpha} 0^{\alpha - (m \bmod \alpha) + 1}$$

and

$$w' = 0^{(\alpha + 1)(m \div \alpha - n \div \alpha)} 1^{n \bmod \alpha} 0^{\alpha - (n \bmod \alpha) + 1} (1^\alpha 0)^{n \div \alpha}.$$

One confirms that both PFAs have the same length and that they are minimal since neither $m \bmod \alpha$ nor $n \bmod \alpha$ is equal to α . Thus, the set $L(A) \cup L(B)$ is accepted by the PFA of the same length which has the set of accepting states that consists of the accepting states of A and B . Therefore the PFA $C = A_{w''}$, for the word $w'' = (1^\alpha 0)^{m \div \alpha + 1}$, accepts $L(A) \cup L(B)$. Clearly, all sequences $1^\alpha 0$ are equivalent while containing no two different equivalent states. So the minimal PFA accepting $L(A) \cup L(B)$ is $A_{w'''}$, for $w''' = 1^\alpha 0$, which has accepting state complexity α . \square

The constructions for the previous lemmata created sequences $1^\alpha 0^\ell$, for some value $\ell \in \mathbb{N}$. There are values for n , m , and α such that α cannot be reached by this method. Therefore we have to create sequences which contain α accepting and distinguishable states which are not consecutive. We want to mention here that we count the positions of the states in an unary PFA in the same way as we count the positions of the letters describing the PFA, namely we start by zero.

Theorem 3.9. *For $m, n \geq 1$ we have*

$$[\min\{m, n\} + 1, \max\{m, n\} - 1] \subseteq g_{\cup, PFA}^{asc, u}(m, n).$$

Proof. Again due to symmetrical reasons we prove the statement for $m \geq n$. We split this proof with regard to the size of n , namely whether it is equal to one or two or it is at least equal to three. Additionally we differentiate between the parity of α in the case $n = 2$ and we prove the case $\alpha = n + 1$ separately for all n . Nevertheless A is minimal in all cases due to an accepting state on an even position, mostly position zero.

1. Case $n = 1$: Since the case $\alpha = n + 1$ will be handled separately we assume that $\alpha \geq 3$. Let $A = A_w$ and $B = A_{w'}$, for $w = 1^2 0^{2(\alpha - 2)} (010^{2(\alpha - 2)})^{m - 2}$ and $w' = 10$, respectively.³ We observe that the DFA accepting the language $L(A) \cup L(B)$ has length $(m - 1) \cdot (\alpha - 1) \cdot 2$ and consists of $m - 1$ equivalent sequences $11(10)^{\alpha - 2}$ each of them containing α accepting states.
2. Case $n = 2$: We define A and B depending on the parity of α .
 - (a) For odd $\alpha > 3$ we set $A = A_w$ and $B = A_{w'}$ for

$$w = (10100^{4(\alpha \div 2 - 1)})(00100^{4(\alpha \div 2 - 1)})^{m - 2} \text{ and } w' = 1100,$$

respectively. Indeed by replicating B we obtain that the DFA accepting $L(A) \cup L(B)$ consists of equivalent sequences $1110(1100)^{\alpha \div 2 - 1}$ each containing α accepting states but no equivalent states.

³This is possible since $n < \min\{m, n\} + 1 \leq \alpha \leq \max\{m, n\} - 1 < m$.

(b) For even $\alpha > 2$ we set $A = A_w$ and $B = A_{w'}$, for

$$w = (1011100^{5(\alpha \div 2 - 2)})(0011100^{5(\alpha \div 2 - 2)})^{m \div 2 - 1} \text{ and } w' = 110^3.$$

If m is even we exchange w by

$$\tilde{w} = (1111100^{5(\alpha \div 2 - 2)})(0011100^{5(\alpha \div 2 - 2)})^{m \div 2 - 2}.$$

Again by replicating the finite state automaton B we obtain that the DFA accepting $L(A) \cup L(B)$ consists of equivalent sequences $1^4 0(110^3)^{\alpha \div 2 - 2}$ each containing α accepting states but no equivalent states.

3. Case $n \geq 3$: Let $B = A_{w'}$, for $w' = 00(10)^n$, and A be the PFA we obtain from A_w , for w being the concatenation of

$$(00)^{n - (\alpha \bmod n) + 1} 11(01)^{\alpha \bmod n - 1} (00)^{(n+1)(\alpha \div n - 1)}$$

and

$$((00)^{n - (\alpha \bmod n) + 1} (01)^{\alpha \bmod n} (00)^{(n+1)(\alpha \div n - 1)})^{m \div \alpha},$$

by adding enough states on even positions which are modulo $2n + 2$ unequal zero to the set of accepting states of A . By letting the positions of those states be as small as possible we obtain the minimality of A from the accepting state on position $2(n - (\alpha \bmod n) + 1)$. If n is a divisor of α we exchange w by

$$\tilde{w} = (01)^{n-1} 1100(00)^{(n+1)(\alpha \div n - 2)} ((01)^n 00(00)^{(n+1)(\alpha \div n - 2)})^{m \div \alpha}.$$

Nevertheless the PFAs A and B are chosen such that the DFA accepting $L(A) \cup L(B)$ consists of $m \div \alpha + 1$ sequences

$$00(10)^{n - (\alpha \bmod n)} (11)^{\alpha \bmod n} (00(01)^n)^{\alpha \div n - 1}$$

and

$$(01(11)^{n-1} 10(00(01)^n)^{\alpha \div n - 2},$$

for $\alpha \bmod n = 0$, respectively, which contain $1 + (2\alpha \bmod n) - 2 + n - (\alpha \bmod n) + 1 + n(\alpha \div n - 1) = \alpha$ accepting states. Those sequences are equivalent while containing no equivalent states.

Therefore we are left to the argumentation that the construction for A is valid regarding to the number m , *i.e.*, we have to prove that the number of accepting states of the DFA accepting $L(A) \cup L(B)$ is strictly greater than m and that m is greater or equal than the number of accepting states we need for A_w ($A_{\tilde{w}}$, respectively). We will restrict our argumentation to the word w because almost the same arguments can be used for \tilde{w} . First we have $(m \div \alpha + 1)\alpha = m - (m \bmod \alpha) + \alpha$ accepting states in the DFA accepting $L(A) \cup L(B)$, which is strictly greater than m .

In A_w we have

$$\alpha \bmod n - 1 + 2 + (\alpha \bmod n)m \div \alpha = 1 + (\alpha \bmod n)(m \div \alpha + 1)$$

accepting states. We define the positive integer z to be $m \div \alpha$ and we observe that $\alpha \bmod n$ is at most equal to $n - 1$. So we need at most $1 + (n - 1)(z + 1) = n(z + 1) - z$ accepting states in A_w . Since $z = m \div \alpha$ we know that m is at least equal to $z\alpha$. Let $\alpha \in [in, (i + 1)n - 1]$, for $i \geq 2$, or $\alpha \in [n + 1, 2n - 1]$.

In the first case $m \geq z\alpha \geq zni > (z+1)n - z$ which is greater or equal to the number of accepting states in A_w .⁴ In the second case we assume that $\alpha = 2n - x$, for $x \in [1, n-1]$. Thus we have $m \geq z\alpha = z(2n - x) = 2nz - xz$. Because $2z$ is strictly greater than $z+1$, for all integers $z \geq 2$, we obtain

$$\begin{aligned} \text{asc}(A_w) &= 1 + (\alpha \bmod n)(m \div \alpha + 1) \\ &= 1 + (2n - x - n)(m \div \alpha + 1) \\ &= 1 + (\alpha - n)(z + 1) = 1 + n(z + 1) - x(z + 1) \end{aligned}$$

which is for $z \geq 2$ less than $2nz - xz \leq m$. For $z = 1$ we also obtain

$$\text{asc}(A_w) = 1 + n(1 + 1) - x(1 + 1) = 2n - 2x + 1 \leq \alpha < m.$$

In conclusion A has at least as many accepting states as needed for A_w , which proves that our construction is valid.

4. Case $\alpha = n + 1$: Let $A = A_w$ and $B = A_{w'}$, for $w = 10^{n-1}10(0^n10)^{m-2}$ and $w' = 1^n00$. We observe that the DFA accepting the set $L(A) \cup L(B)$ which has length $(n+2)(m-1)$ consists of $m-1$ sequences $1^{n+1}0$. Obviously all sequences are equivalent and each of them contains no equivalent states. Every sequence has $n+1$ accepting states which shows the reachability of $\alpha = n+1$.

□

Obviously $K \cup \emptyset = K$ and $\emptyset \cup L = L$, for every languages $K, L \subseteq \Sigma^*$. Together with the Theorems 3.5, 3.8, and 3.9 we obtain the following corollary.

Corollary 3.10. *We have*

$$g_{\cup}^{\text{asc},u}(m, n) = g_{\cup, \text{PFA}}^{\text{asc},u}(m, n) = \begin{cases} \{m\}, & \text{if } n = 0, \\ \{n\}, & \text{if } m = 0, \\ \mathbb{N}, & \text{otherwise.} \end{cases}$$

□

3.4. Difference

Now let us come to the difference operation which was also considered in [3]. For deterministic finite automata with no restrictions the following result was shown:

$$g_{\setminus}^{\text{asc},u}(m, n) = g_{\setminus}^{\text{asc}}(m, n) = \begin{cases} \{m\}, & \text{if } n = 0 \\ \{0\}, & \text{if } m = 0 \\ \{0\} \cup \mathbb{N}, & \text{otherwise,} \end{cases}$$

Again the languages that prove these results are *not* accepted by any PFA.

This subsection is structured as follows. Let $C(L)$ refer to the complement of the language L . First we will use the fact that $K \setminus L = K \cap C(L)$, for all finite languages K and L , in order to show that all numbers in the range $[0, m]$ are reachable and for all $m, n \geq 1$ with $\alpha \bmod m = 0$ the numbers α are reachable as well. The previously mentioned fact for the set difference of two languages allows us to prove the first two statements by constructing A and B such that the minimal DFA accepting the language of the direct product of A and \overline{B} has the required size, for \overline{B} being equal to B except that its set of accepting states is complemented. Afterwards we prove that for all $m, n \geq 1$ with $\alpha \bmod m > 0$ the numbers α are reachable, too.

⁴The inequality $zni > (z+1)n - z$ holds because $i \geq 2$ and $z \geq 1$.

Lemma 3.11. *For $m, n \geq 1$ we have $[0, m] \subseteq g_{\setminus, PFA}^{asc, u}(m, n)$.*

Proof. Let $A = A_w$ and $B = A_{w'}$, for $w = (0^n 1)^\alpha (10^n)^{m-\alpha} 0^{n+1}$ and $w' = 1^n 0^1$. That means $L(A) \setminus L(B)$ is accepted by $A_{w''}$, for $w'' = (0^n 1)^\alpha 0^{(n+1) \cdot (m-\alpha+1)}$. We leave it to the reader to observe that the three involved automata are minimal. The PFA $A_{w''}$ has accepting state complexity α which proves the statement of the lemma. \square

The next lemma shows that the every number α in the upper range $\mathbb{N}_{\geq m+1}$ is obtainable, if α fulfills $\alpha \bmod m = 0$.

Lemma 3.12. *For $m, n \geq 1$ and $\alpha \in \mathbb{N}_{\geq m+1}$ with $\alpha \bmod m = 0$ we have that the value $\alpha \in g_{\setminus, PFA}^{asc, u}(m, n)$.*

Proof. Let $\alpha = mx + (\alpha \bmod m)$. We set $A = A_w$ and $B = A_{w'}$, for the words $w = 1^m 0^k$ and $w' = 0^x 1^n$, respectively, where k is the smallest positive integer such that $\gcd(m+k, x+n) = 1$. Since A and B contain only one continuous sequence of accepting states they are minimal. We observe that the the cross product DFA of A and $\bar{B} = A_{w''}$, for $w'' = 1^x 0^n$, has xm accepting states and is minimal due to the Chinese Remainder Theorem. \square

The upper interval $\mathbb{N}_{\geq m+1}$ is also attainable, for $\alpha \bmod m \neq 0$, which clearly proves the range to be reachable for all numbers in the interval.

Lemma 3.13. *For $m, n \geq 1$ and $\alpha \in \mathbb{N}_{\geq m+1}$ with $\alpha \bmod m \neq 0$ we have that the value $\alpha \in g_{\setminus, PFA}^{asc, u}(m, n)$.*

Proof. Let $A = A_w$ and $B = A_{w'}$, for words $w, w' \in \{0, 1\}^*$. We assume that x_m and x_n are positive integers such that $|w^{x_m}| = |w'^{x_n}|$. It is not hard to see that the automaton C which consists of the tuples⁵ defined by (w^{x_m}, w'^{x_n}) accepts $L(A) \setminus L(B)$. Since the words w and w' are replicated we speak likewise of replications of A and B , too. We observe that in each replication of A each state of B matches either $\lceil x_n/x_m \rceil$ or $\lfloor x_n/x_m \rfloor$ states of A . Since we have already proven the reachability for $\alpha \bmod m = 0$ we assume for the rest of the proof that $\alpha \bmod m$ is at least equal to one. We choose $x_m = 1 + \alpha \div m$ and x_n such that $\lceil x_n/x_m \rceil = m - (\alpha \bmod m)$ and $\lfloor x_n/x_m \rfloor \neq m - (\alpha \bmod m)$. This means we choose the lengths of A and B appropriately to obtain those numbers of replications and that A and B contain more than m and n states, respectively. Clearly, we can increase the lengths of A and B while keeping their ratio, *i.e.*, their multipliers so it is possible to choose the accepting states of A such that exactly $m - (\alpha \bmod m)$ accepting states of the first replication of A are matched by a specific accepting state p of B and its replications. It is not hard to see that if we would restrict A to contain exactly those $m - (\alpha \bmod m)$ accepting states there would be no other replication of A having an accepting state which matches with p in any other replication of B . We will argue later why we can choose the remaining accepting states of A and B such that they do not match an accepting state in any replication of B and A , respectively.

We observe that an increase of the lengths of A and B , while keeping their ratio, let the density of accepting states tend to zero. On the other side we see that each of the m accepting states of A appear only in x_m states as a first component, which means there are only $x_m \cdot m$ states which have an accepting state as their first component. Therefore the above described length increase of A and B ensures that on the one hand there are states in A which are not the first component of an initially reachable state in C , whose second component is accepting and on the other hand there are states in B which are not the second component of an initially reachable state in C whose first component is accepting. Clearly, the number of those states increases if the length of A and B increases. If we choose the lengths large enough it is easy to see that we can choose the positions of $\alpha \bmod m$ accepting states of A and $n - 1$ accepting states of B such that none of those states is a first or second component, respectively, of an initially reachable state in the product automaton that has two accepting components.

It remains to show that A and B can be chosen with the above described procedure such that A , B , and C are minimal. Since $\alpha \bmod m \neq 0$ there is an accepting state of A whose position can be chosen such that the

⁵A state is accepting if the first component is accepting and the second one is not accepting.

requirements of Lemma 3.2 are fulfilled. For this it may be necessary to increase the length of A and B again but it is obvious that if the length tends to infinity we can choose the position of exactly one accepting state of A such that its distance to its foregoing accepting state is proportional to the length of A . Clearly, we can request the analogue for B since we only used one of its accepting states to match the required number of accepting states of A .⁶ So Lemma 3.2 implies the minimality of A and B . Since only accepting states in the first replication of A are matched by accepting states of B this also implies that C is minimal, too. \square

By combining the Lemmata 3.11, 3.12, and 3.13 we obtain the following corollary.

Corollary 3.14. *We have*

$$g_{\setminus}^{\text{asc},u}(m, n) = g_{\setminus, PFA}^{\text{asc},u}(m, n) = \begin{cases} \{m\}, & \text{if } n = 0, \\ \{0\}, & \text{if } m = 0, \\ \{0\} \cup \mathbb{N}, & \text{otherwise.} \end{cases}$$

Proof. Since $K \setminus \emptyset = K$ and $\emptyset \setminus L = \emptyset$, for every languages $K, L \subseteq \Sigma^*$, the first two statements follow immediately. Additionally the last statement follows by the Lemmata 3.11, 3.12, and 3.13. \square

3.5. Intersection

We show that the left open unary case for the intersection operation for both PFAs and DFAs differs from the solved general case. It is not hard to see that at most the numbers in the range $[0, nm]$ can be reached. We split this interval into three smaller ones, namely $[0, \max\{m, n\}]$, $[\max\{m, n\} + 1, nm - \min\{m, n\}]$, and $[nm - \min\{m, n\} + 1, nm]$.

Lemma 3.15. *We have $[0, \max\{m, n\}] \subseteq g_{\cap, PFA}^{\text{asc},u}(m, n)$.*

Proof. Since for the intersection operation the ordering of the input languages is irrelevant we assume $m \geq n$. Let $A = A_w$ and $B = A_{w'}$, for $w = (10^n)^\alpha (0^n 1)^{m-\alpha} 0^{n+1}$ and $w' = 1^n 0$, respectively. Both PFAs are minimal since the last $n + 1$ states of A do not contain an accepting state and the accepting states of B form a sequence. The minimal DFA accepting the language $L(A) \cap L(B)$ is the PFA $A_{w''}$, for $w'' = (10^n)^\alpha 0^{(n+1) \cdot (m-\alpha+1)}$, which obviously has accepting state complexity α . \square

Concerning the question which numbers are magic, we exhaustively searched for these numbers for small m and n . We think that from the obtained data we can support the following general conjecture:

Conjecture 3.16. *All numbers in $[\max\{m, n\} + 1, nm - \min\{m, n\}]$ which are not in*

$$\begin{aligned} & [\max\{m, n\}, n + m] \\ & \cup \{t_n x_m \mid t_n \text{ is a nonzero divisor of } n \text{ and } 0 \leq x_m \leq (nm - \min\{m, n\}) \div t_n\} \\ & \cup \{t_m x_n \mid t_m \text{ is a nonzero divisor of } m \text{ and } 0 \leq x_n \leq (nm - \min\{m, n\}) \div t_m\} \end{aligned}$$

are magic for the intersection of p -regular languages.

Next we investigate the numbers in the range $[nm - \min\{m, n\} + 1, nm]$. For showing that all numbers except nm are magic we prove the following structural property of the cross product for PFAs.

Lemma 3.17. *Let q, q' and p, p' be arbitrary states of the minimal unary PFAs A and B , respectively. If (q, p) , (q', p) and (q, p') are initially reachable in the cross product automaton C then (q', p') is initially reachable, too.*

Proof. Let $\Sigma = \{a\}$ be the input alphabet of A, B , and C . Since (q, p) , (q', p) , and (q, p') are initially reachable in C we know that there are words w_q and w_p which map (q, p) onto (q', p) and (q, p) onto (q, p') . Because A

⁶For $n = 1$ the PFA B is trivially minimal.

and B are unary we observe that w_q and w_p induce the identity on B and A , respectively. This implies that $(q, p) \cdot w_q w_p = (q', p) \cdot w_p = (q', p')$ which proves the stated claim. \square

One may ask whether Lemma 3.17 holds for alphabets of arbitrary size. In general it is not true that w_q and w_p induce the identity on B and A , respectively. Instead those words induce a cycle on B and A that has a size that divides the order of the word. If the cycle of A and B contain q, q' and p, p' , respectively, the statement of the lemma remains true. We conjecture that the lemma above does not hold for alphabets of at least two letters. As mentioned before we use Lemma 3.17 to prove that a whole range of numbers in the upper interval cannot be reached.

Theorem 3.18. *We have $[nm - \min\{m, n\} + 1, nm - 1] \not\subseteq g_{\cap, PFA}^{\text{asc}, u}(m, n)$.*

Proof. Let $\alpha \in [nm - \min\{m, n\} + 1, nm - 1]$. Clearly, Lemma 3.17 implies that for all PFAs A and B their cross product automaton has either less than $nm - \min\{m, n\} + 1$ or nm initially reachable accepting states. On the other hand a result from [6], Lemma 4 implies that if a PFA has nm accepting states and it is *not* minimal, then its minimal DFA has t accepting states for a divisor t of nm . Since every divisor of nm is less than $nm - \min\{m, n\} + 1$ the claim of the theorem follows. \square

If we look at the cross product automaton C of unary DFAs A and B we see that for every state q that is in the tail of A or B there is exactly one initially reachable state in C which contains q as one of its components. So we obtain that C contains at most $nm - \min\{m, n\} + 1$ initially reachable accepting states if A or B has an accepting state in their tail. Together with Theorem 3.18 we obtain the following corollary.

Corollary 3.19. *We have $[nm - \min\{m, n\} + 2, nm - 1] \not\subseteq g_{\cap}^{\text{asc}, u}(m, n)$.* \square

As mentioned before the upper bound for $g_{\cap, PFA}^{\text{asc}, u}(m, n)$ is *not* a magic number which is proven in the following lemma.

Lemma 3.20. *We have $nm \in g_{\cap, PFA}^{\text{asc}, u}(m, n)$.*

Proof. Let $A = A_w$ and $B = A_{w'}$, for $w = 1^m 0^n$ and $1^n 0^{m+1}$, respectively. Since $n + m$ and $n + m + 1$ are coprime it is obvious that the length of their product automaton C is $(n + m) \cdot (n + m + 1)$. So each pair of accepting states is initially reachable. We observe that there are $\max\{m, n\} - \min\{m, n\} + 1$ sequences of accepting states of length $\min\{m, n\}$. All of those sequences follow each other, *i.e.*, only non-accepting states are between them. There are also shorter sequences of accepting states in C , *e.g.*, an accepting state which follows and is followed by a non-accepting state. This implies that C is minimal which proves the stated claim. \square

3.6. Reversal

The results of this subsection are in contrast to the general case, where arbitrary DFAs are considered. Here the restriction for the input automaton to be a PFA provides magic numbers which are not magic if the input automaton is not restricted. For deterministic finite automata with no restrictions the following result was proven in [7], Theorem 12:

$$g_R^{\text{asc}}(m, n) = \begin{cases} \{0\}, & \text{if } m = 0 \\ \mathbb{N}, & \text{otherwise,} \end{cases}$$

We show that in the case of permutation automata the number $\alpha = 1$ is magic for all $m \geq 2$ and $\alpha \geq 2$ is magic for $m = 1$. Before we do this we need a special DFA that plays an important role for the reversal operation. We want to mention that for a unary language L its reversal L^R is equal to L . So we will only consider languages with at least two different letters. First we define $\binom{S}{k}$, for a finite set S and a non-negative integer k , to be the

set of all subsets of S which have size k . For a PFA $A = (\{q_0, q_1, \dots, q_{k-1}\}, \Sigma, \cdot_A, q_0, F_A)$, we define

$$A_R = \left(\binom{Q_A}{|F_A|}, \Sigma, \cdot_{A_R}, F_A, \{R \in \binom{Q_A}{|F_A|} \mid q_0 \in R\} \right),$$

where $R \cdot_{A_R} w = \{q \cdot_A w^{-1} \mid q \in R, w \in \Sigma\}$, for all $R \in \binom{Q_A}{|F_A|}$. We want to mention here that A_R is a well-defined DFA since for every word w the mapping w^{-1} is uniquely defined because A is a PFA. Because w^{-1} applies the reverse transitions to every state of A , and every state of A_R that contains the initial state of A is accepting, it is not hard to see that A_R accepts the language $L(A)^R$. Before we prove our results for the accepting state complexity of the reversal operation of PFAs we derive two structural properties of the DFA A_R . First we count the number of initially reachable states in A_R .

Lemma 3.21. *Let A be a minimal PFA. Then there is an integer $x \geq 1$ such that for every state q of A there are x initially reachable states in A_R containing q .*

Proof. Let q be an arbitrary state of A . Assume there are $x \geq 1$ states R_0, R_1, \dots, R_{x-1} in A_R which contain q . Since A is a PFA the images of those states are different regardless of the choice of the mapping. If we apply the mapping which maps q onto q' , for any other state q' of A , it follows directly that there are at least x states of A_R which contain q' . Since this argument can be used symmetrically to the chosen states q and q' the claim of the lemma follows. \square

Since the states of A_R are in turn sets we prove the following property of A_R which is the automata-theoretical interpretation of the fact that bijections on elements induce bijections on sets of those elements.

Lemma 3.22. *For every PFA A the DFA A_R is a PFA, too.*

Proof. Since A is a PFA for every letter a the preimage of any state q of A is uniquely defined. By applying this property to every state q in a state R of A_R we directly obtain the unique preimage of R . \square

Now we will prove our magic number result for the PFA case of the reversal operation.

Lemma 3.23. *Let $m \geq 2$. Then there exists no PFA A with $\text{asc}(A) = m$ such that $\text{asc}(A_R) = 1$.*

Proof. Since $L(A_R)^R = (L(A)^R)^R = L(A)$ we can formulate the following statement which is equivalent to the one of the lemma to prove:

Let $m \geq 2$. Then there exists *no* PFA A with $\text{asc}(A) = 1$ such that $\text{asc}(A_R) = m$.

Indeed it is easy to confirm that if A has only one accepting state the definition of A_R directly implies that A_R has only one accepting state, too. This proves the stated claim. \square

The result of the previous lemma proves the inequality statement of our main theorem for the accepting state complexity of the reversal operation of p-regular languages. Obviously we also prove that for m equal to two every number unequal to one is *not* magic. We do this by constructing an automaton A such that A_R has $\alpha \cdot k \div m = \binom{k}{m}$ initially reachable states while every state of A appears in exactly α of them.

Theorem 3.24. *We have*

$$g_{R,PFA}^{\text{asc}}(m) = \begin{cases} \{0\}, & \text{if } m = 0, \\ \{1\}, & \text{if } m = 1, \\ \mathbb{N}_{\geq 2}, & \text{if } m = 2, \end{cases}$$

and $g_{R,PFA}^{\text{asc}}(m) \neq \mathbb{N}$, if $m \geq 3$. Therefore $g_{R,PFA}^{\text{asc}}(m) \neq g_R^{\text{asc}}(m)$.

Proof. Let A be a minimal PFA with $\text{asc}(A) = m$. For a word w , by w^{-1} we denote the word $w^{o(w)-1}$, where $o(w)$ is the order of w , *i.e.*, the smallest number such that $w^{o(w)}$ induces the identity. Clearly, this is well-defined since we are dealing with PFAs. The first two statements follow directly from the construction of A_R . Due to Lemma 3.23 it only remains to show for $m = 2$ that the numbers in $\mathbb{N}_{\geq 2}$ are reachable. We want to construct A_R such that exactly α inequivalent accepting states are initially reachable. First we observe that due to Lemma 3.21 this results in $\alpha \cdot k \div 2$ initially reachable states in A_R in total or some multiple of it, where k is the number of states of A . Moreover, we see that $\alpha \cdot k \div 2 = \binom{k}{2}$ has always an integer solution that is greater than two, namely it is equivalent to $k = \alpha + 1$. Indeed $\binom{k}{2}$ is the number of states of A_R which means that all states of A_R are initially reachable. We define $A = (\{q_0, q_1, \dots, q_\alpha\}, \{a, b\}, \cdot_A, q_0, \{q_0, q_1\})$, where

$$\begin{aligned} q_i \cdot_A a &= q_{i+1 \bmod (\alpha+1)}, & \text{for } 0 \leq i \leq \alpha, \\ q_i \cdot_A b &= q_{i+1 \bmod 2}, & \text{for } 0 \leq i \leq 1, \end{aligned}$$

and all other transitions are self-loops. Obviously, the letter a induces a permutation of order $\alpha + 1 = k$ and b induces a transposition. A basic result from group theory states that the mappings induced by a and b generate the symmetric group on k elements. This can be seen by observing that the mappings induced by the words in $\Sigma_G = \{a^i b (a^i)^{-1} \mid 0 \leq i \leq k - 1\}$ form the set of transpositions $\{(q_i, q_{i+1 \bmod k}) \mid 0 \leq i \leq k - 1\}$, which is obviously a generating set from the symmetric group on k elements. Since the symmetric group on k elements is transitive on the set of subsets of size 2 it follows that every state in A_R is initially reachable, see for example [1].

It remains to show that A and A_R are minimal PFAs to provide the statement of the theorem. For A this follows directly from the fact that the symmetric group contains all possible transpositions and these fix all states except two.⁷ Especially every pair of states can be mapped onto every other pair of states, *i.e.*, a pair of an accepting and a non-accepting state. We see that b maps every state containing q_0 onto a state not containing q_0 except $\{q_0, q_1\}$. Since containment of q_0 is the condition for a state to be accepting we see that no state can be equivalent to $\{q_0, q_1\}$. By Lemma 3.22 the automaton A_R is a PFA. Next, by Lemma 14 of [6], all initially reachable states of A_R are equivalent to the same number of states. Therefore, no different pair of states of A_R can be equivalent to each other, which implies that A_R is minimal. \square

We note here that for $m \geq 3$ the equation $\alpha \cdot k \div m = \binom{k}{m}$ has no integer solution for many values of α and m which can be easily confirmed. For those values of α and m , for which the equation has an integer solution, we obtain $\alpha \in g_{R, \text{PFA}}^{\text{asc}}(m)$ in similar fashion like for $m = 2$. Nevertheless we conjecture the following:

Conjecture 3.25. *We have*

$$g_{R, \text{PFA}}^{\text{asc}}(m) = \begin{cases} \{0\}, & \text{if } m = 0, \\ \{1\}, & \text{if } m = 1, \\ \mathbb{N}_{\geq 2}, & \text{if } m \geq 2, \end{cases}$$

Clearly, this would mean that $\alpha = 1$ is the only number which is magic for the reversal of p-regular languages and non-magic for arbitrary regular languages.

3.7. Quotient

For two DFAs A and B the right quotient $L(A)L(B)^{-1}$ can be accepted by the DFA \tilde{A} which can be obtained from A by exchanging its set of accepting states F by $\{q \mid q \cdot w \in F \text{ for some } w \text{ in } L(B)\}$, which we denote by \tilde{F} . It is obvious that \tilde{A} is a PFA if A is a PFA. Additionally, if q_0 is the initial state of A , then the automaton obtained from A , by making all states in $\{q_0 \cdot w \mid w \in L\}$ initial, accepts $L(B)^{-1}L(A)$. Since for unary languages

⁷With the previous mentioned generating set Σ_G it is obvious that there is a word for each mapping of the symmetric group.

the left and right quotient coincide no distinction is made at this point and we use the right quotient unless otherwise stated. For regular languages in general the following result was shown in [7], Theorem 10:

$$g_{-1}^{\text{asc},u}(m, n) = g_{-1}^{\text{asc}}(m, n) = \begin{cases} \{0\}, & \text{if } m = 0 \text{ or } n = 0, \\ \{0\} \cup \mathbb{N}, & \text{otherwise,} \end{cases}$$

Clearly, the first statement follows directly from the fact that $K\emptyset^{-1} = \emptyset L^{-1} = \emptyset$, for all languages K and L . But we show that the last statement does not hold for the class of p-regular languages. For this we distinguish whether n is equal to one or at least equal to two. First we show which numbers are reachable if n is equal to one.

Lemma 3.26. *We have $[1, m] \subseteq g_{-1, \text{PFA}}^{\text{asc},u}(m, 1)$.*

Proof. Let α be in $[1, m]$. Define $A = A_w$ and $B = A_{w'}$, for

$$w = 1^\alpha 0^{m+1-\alpha} (10^m)^{m-\alpha} 0^{m+1} \quad \text{and} \quad w' = 010^{m-1}.$$

We observe that A has length $(m+1)(m-\alpha+2)$ and B has length $m+1$. It is not hard to see that

$$\begin{aligned} L(A) = \{ & a^{i+x(m+1)(m-\alpha+2)} \mid 0 \leq i \leq \alpha-1 \text{ and } 0 \leq x \} \\ & \cup \{ a^{(m+1)i+x(m+1)(m-\alpha+2)} \mid 1 \leq i \leq m-\alpha \text{ and } 0 \leq x \} \end{aligned}$$

and

$$L(B) = \{ a^{(m+1)i+1} \mid i \in \mathbb{N} \cup \{0\} \},$$

for a being the letter of the input alphabet of A and B . We observe that the PFA \tilde{A} has the set of accepting states \tilde{F} that contains exactly the elements $q_{(i-((m+1)j+1)) \bmod (m+1)(m-\alpha+2)}$, for $0 \leq i \leq \alpha-1$ or $i \in \{(m+1)\ell \mid 1 \leq \ell \leq m-\alpha\}$ and $j \in \mathbb{N} \cup \{0\}$. Alternatively we can write

$$\tilde{F} = \{ q_{(i-((m+1)j+1)) \bmod (m+1)(m-\alpha+2)} \mid 0 \leq i \leq \alpha-1 \text{ and } j \in \mathbb{N} \cup \{0\} \}$$

because

$$\begin{aligned} 0 - ((m+1)(m-\alpha+2-j) + 1) \bmod (m+1)(m-\alpha+2) \\ = (m+1)j - ((m+1) \cdot 0) + 1 \bmod (m+1)(m-\alpha+2) \end{aligned}$$

holds. One may observe that \tilde{F} contains all accepting states of A but their index is decreased by one modulo the length of A . If we shift those states again by an arbitrary multiple of $(m+1)$ we obtain the remaining states in \tilde{F} . Clearly, \tilde{F} does not contain other states. Therefore automaton $\tilde{A} = A_{w''}$, for the word $w'' = 1^{\alpha-1} 0^{m+1-\alpha} (1^\alpha 0^{m+1-\alpha})^{m-\alpha+1} 1$. Indeed this PFA is not minimal, *i.e.*, all of its sequences $1^{\alpha-1} 0^{m+1-\alpha} 1$ are equivalent. Thus the minimal PFA accepting the language $L(A)L(B)^{-1}$ is $A_{w'''}$, for the word $w''' = 1^{\alpha-1} 0^{m+1-\alpha} 1$ which has accepting state complexity α . \square

Next we prove that every number which is *not* reachable due to the previous lemma is magic.

Lemma 3.27. *We have $[1, m] = g_{-1, \text{PFA}}^{\text{asc},u}(m, 1)$.*

Proof. Due to the proof of Lemma 3.26 it remains to show that $\mathbb{N}_{\geq m+1}$ is not in $g_{-1, \text{PFA}}^{\text{asc},u}(m, 1)$. Therefore let A and B be unary minimal PFAs with m and n accepting states, respectively. Recall that \tilde{A} is the PFA obtained

from A by replacing its set of accepting states with $\tilde{F} = \{q \mid q \cdot w \in F \text{ for some } w \text{ in } L(B)\}$. We observe that the set of accepting states of \tilde{A} is equal to

$$\tilde{F} = \{q_{(i-(jk'+\ell)) \bmod k} \mid i \in I_A \text{ and } j \in \mathbb{N} \cup \{0\}\}$$

for I_A being the index set of the accepting states of A , q_ℓ being the accepting state of B and k, k' being the number of states of A and B , respectively. For an arbitrary but fixed $i \in I_A$ we see that each of the states

$$q_{(i-(0k'+\ell)) \bmod k}, q_{(i-(1k'+\ell)) \bmod k}, q_{(i-(2k'+\ell)) \bmod k}, \dots$$

can be mapped by $a^{k'}$ onto its predecessor. Since this holds for every $i \in I_A$ those states have to be equivalent which proves that \tilde{A} contains at most m inequivalent accepting states. \square

Now we generalize Lemma 3.26, for $n \geq 2$.

Lemma 3.28. *We have $[1, mn] \subseteq g_{-1, PFA}^{\text{asc}, u}(m, n)$, for $n \geq 2$.*

Proof. Let $\alpha = nx + y$, for an integer $1 \leq x \leq m - 1$ and $0 \leq y \leq n$. We argue in a similar way like in the proof of Lemma 3.26. Therefore we will reuse \tilde{A} and \tilde{F} from there. Roughly speaking the automaton B will be constructed such that the accepting states of \tilde{A} are obtained by decreasing the positions of the accepting states of A by $0, 1, \dots, n - 1$. Clearly, this results in

$$\tilde{F} = \{q_{(i-(jk'+\ell)) \bmod k} \mid i \in I_A, j \in \mathbb{N} \cup \{0\}, \text{ and } \ell \in \{0, 1, \dots, n - 1\}\},$$

where I_A is the index set from the proof of Lemma 3.27. By choosing the length of A, B and the positions of the accepting states of A accordingly we obtain α equivalence classes of accepting states in \tilde{A} . We define the automata $A = A_w$ and $B = A_{w'}$, for

$$w = \begin{cases} (10^{n-1})^{x-1} 10^{y-1} 10^n (10^{nx+y})^{m-(x+1)} 0^{nx+y+1}, & \text{if } y \neq 0, \\ (10^{n-1})^{x-1} 10^n (10^{nx})^{m-(x+1)} 0^{nx+1}, & \text{if } y = 0, \end{cases}$$

and

$$w' = 1^n 0^{n(x-1)+y+1},$$

respectively. Obviously A has length

$$\begin{aligned} (1+n-1)(x-1) + 1 + (y-1) + 1 + n + (1+(nx+y))(m-(x+1)) + (nx+y+1) \\ = (nx+y+1)(m-x+1), \end{aligned}$$

for $y \neq 0$,

$$\begin{aligned} (1+n-1)(x-1) + 1 + n + (nx+1)(m-(x+1)) + nx + 1 \\ = (nx+1)(m-x+1) = (nx+y+1)(m-x+1) \end{aligned}$$

for $y = 0$, and B has length $nx + y + 1$. In both cases A is minimal since it contains a sequence of non-accepting states of length $nx + y + 1$ which is greater than the shortest word that maps two accepting states onto each other which only have non-accepting states between them. It is obvious that B is also minimal. Since the arguments in both cases, $y \neq 0$ and $y = 0$, are quite similar we will limit the proof to the case $y \neq 0$. It is not

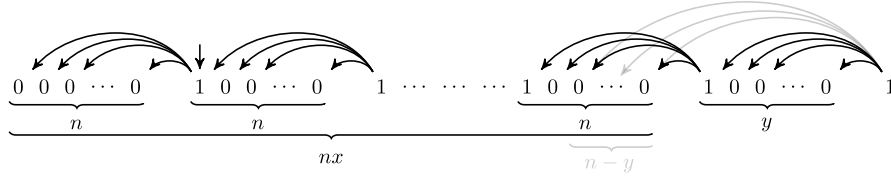


FIGURE 1. Each accepting state of A is shifted by the accepting states of B onto its n predecessors. This is shown for the first $n(x-1) + y + 1$ states of A such that an arrow from an accepting state q , encoded by a 1, to a state q' , encoded by a number, means that instead of q being accepting the state q' is accepting in \tilde{A} . Additionally the vertical arrow marks the starting state of A .

hard to see that

$$L(B) = \bigcup_{\ell \in \{0, 1, \dots, n-1\}} \{a^{(n(x-1)+y+1)i+\ell} \mid i \in \mathbb{N} \cup \{0\}\},$$

for a being the letter of the input alphabet of A and B . With the construction in the proof of Lemma 3.26 we see that B induces the initially described shifts on the positions of the accepting states of A for obtaining the positions of the accepting states of \tilde{A} . We observe that the PFA \tilde{A} has the set of accepting states which is equal to

$$\begin{aligned} \tilde{F} = & \bigcup_{\ell \in \{0, 1, \dots, n-1\}} \{q_{(ni - ((nx+y+1)j+\ell)) \bmod (nx+y+1)(m-x+1)} \mid 0 \leq i \leq x-1 \text{ and} \\ & j \in \mathbb{N} \cup \{0\}\} \\ & \cup \{q_{(nx+y - ((nx+y+1)j+\ell)) \bmod (nx+y+1)(m-x+1)} \mid j \in \mathbb{N} \cup \{0\}\} \end{aligned}$$

because of a similar argument we used in the proof of Lemma 3.26. This can be seen by looking at the way the accepting states of A are shifted to obtain \tilde{A} which is shown for the first $n(x-1) + y + 1$ states of A in Figure 1.

Therefore $\tilde{A} = A_{w''}$, for

$$w'' = 1^{n(x-1)+y+1}0(1^{nx+y}0)^{m-x}1^{n-1}.$$

It is not hard to deduce that all sequences $1^{n(x-1)+y+1}01^{n-1}$ are equivalent in \tilde{A} while each of those sequences does not contain equivalent states. Therefore, the minimal PFA accepting the language $L(A)L(B)^{-1}$ is $A_{w'''}$, for the word $w''' = 1^{n(x-1)+y+1}01^{n-1}$, which has $n(x-1) + y + 1 + n - 1 = nx + y = \alpha$ accepting states. \square

Next we rule out every number that is not reachable by Lemma 3.28. Like in the proof of Lemma 3.27 one observes that the set of accepting states of the DFA accepting the quotient language of two p-regular languages L_1 and L_2 of accepting state complexity m and n is given by applying the following two steps. First the accepting states of the minimal DFA accepting L_1 are shifted to n positions. Afterwards these mn accepting states are cyclically replicated by the length of the DFA accepting L_1 . Since the DFA accepting the quotient language is a PFA all cyclic replications are equivalent.

Lemma 3.29. *We have $[1, mn] = g_{-1, PFA}^{\text{asc}, u}(m, n)$.*

Proof. Recall the notation from the beginning of the subsection for the quotient operation. For the set of accepting states \tilde{F} of \tilde{A} we see that

$$\tilde{F} = \{q_{(i-(jk'+\ell)) \bmod k} \mid i \in I_A, j \in \mathbb{N} \cup \{0\}, \text{ and } \ell \in I_B\} = \bigcup_{\ell \in I_B} \{q_{(i-(jk'+\ell)) \bmod k} \mid i \in I_A \text{ and } j \in \mathbb{N} \cup \{0\}\}$$

holds, if I_A and I_B are the sets of indices of the accepting states of A and B , respectively. Due to Lemma 3.27 each set of the union contains at most m inequivalent accepting states. Since $|I_B| = n$ holds we obtain that the number of inequivalent accepting states is at most mn . \square

By using the previous four Lemmata 3.26, 3.27, 3.28, and 3.29 we deduce the following corollary.

Corollary 3.30. *We have*

$$g_{-1, PFA}^{\text{asc}, u}(m, n) = \begin{cases} \{0\}, & \text{if } m = 0 \text{ or } n = 0, \\ [1, mn], & \text{otherwise.} \end{cases}$$

Therefore $g_{-1, PFA}^{\text{asc}, u}(m, n) \neq g_{-1}^{\text{asc}, u}(m, n)$. \square

The accepting state complexity for the quotient operation on languages accepted by permutation automata with larger input alphabets has to be left open and is subject to further research.

Acknowledgements. We thank Michal Hospodár and the anonymous referees for the helpful suggestions and valuable comments. This paper is a completely revised and expanded version of a paper presented at the 12th International Workshop on Non-Classical Models of Automata and Applications (NCMA) held in Debrecen, Hungary, August 26–27, 2022.

REFERENCES

- [1] R.A. Beaumont and R.P. Peterson, Set-transitive permutation groups. *Can. J. Math* **7** (1955) 35–42.
- [2] J. Brzozowski, Open problems about regular languages, edited by R.V. Book. *Formal Language Theory*. Academic Press (1980) 23–47.
- [3] J. Dassow, On the number of accepting states of finite automata. *J. Autom. Lang. Comb.* **21** (2016) 55–67.
- [4] Y. Gao, N. Moreira, R. Reis and S. Yu, A survey on operational state complexity. *J. Autom. Lang. Comb.* **21** (2016) 251–310.
- [5] M.A. Harrison, *Introduction to Formal Language Theory*. Addison-Wesley (1978).
- [6] M. Holzer and C. Rauch, The range of state complexities of languages resulting from the cascade product – the unary case, edited by S. Maneth. *Proceedings of the 25th International Conference on Implementation and Application of Automata*. Vol. 12803 of *LNCS*. Bremen, Germany. Springer (2021) 90–101.
- [7] M. Hospodár and M. Holzer, The ranges of accepting state complexities of languages resulting from some operations, in edited by C. Campeanu. *Proceedings of the 23th Conference on Implementation and Application of Automata*. Vol. 10977 of *LNCS*. Charlottetown, Prince Edward Island, Canada. Springer. (2018) 198–210.
- [8] M. Hospodár and P. Mlynárčik, Operations on permutation automata, edited by N. Jonoska and D. Savchuk. *Proceedings of the 24th International Conference on Developments in Language Theory*. Vol. 12086 of *LNCS*. Tampa, Florida, USA. Springer (2020) 122–136.
- [9] K. Iwama, Y. Kambayashi and K. Takaki, Tight bounds on the number of states of DFAs that are equivalent to n -state NFAs. *Theoret. Comput. Sci.* **237** (2000) 485–494.
- [10] I. Jecker, N. Mazzocchi and P. Wolf, Decomposing permutation automata, edited by S. Haddad and D. Varacca. *Proceedings of the 32nd International Conference on Concurrency Theory*. Vol. 203 of *LIPICs*. Virtual Conference. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, Dagstuhl, Germany (2021) 18:1–18:19.
- [11] H.P. Zeiger, Yet another proof of the cascade decomposition theorem for finite automata. *Math. Syst. Theory* **1** (1967) 225–228.



Please help to maintain this journal in open access!

This journal is currently published in open access under the Subscribe to Open model (S2O). We are thankful to our subscribers and supporters for making it possible to publish this journal in open access in the current year, free of charge for authors and readers.

Check with your library that it subscribes to the journal, or consider making a personal donation to the S2O programme by contacting subscribers@edpsciences.org.

More information, including a list of supporters and financial transparency reports, is available at <https://edpsciences.org/en/subscribe-to-open-s2o>.