Most Complex Non-Returning Regular Languages⋆

Janusz A. Brzozowski[1] and Sylvie Davies[2]

[1] David R. Cheriton School of Computer Science, University of Waterloo
Waterloo, ON, Canada N2L 3G1
brzozo@uwaterloo.ca
[2] Department of Pure Mathematics, University of Waterloo
Waterloo, ON, Canada N2L 3G1
sldavies@uwaterloo.ca

Abstract. A regular language L is non-returning if in the minimal deterministic finite automaton accepting it there are no transitions into the initial state. Eom, Han and Jirásková derived upper bounds on the state complexity of boolean operations and Kleene star, and proved that these bounds are tight using two different binary witnesses. They derived upper bounds for concatenation and reversal using three different ternary witnesses. These five witnesses use a total of six different transformations. We show that for each $n \geqslant 4$ there exists a ternary witness of state complexity n that meets the bound for reversal and that at least three letters are needed to meet this bound. Moreover, the restrictions of this witness to binary alphabets meet the bounds for product, star, and boolean operations. We also derive tight upper bounds on the state complexity of binary operations that take arguments with different alphabets. We prove that the maximal syntactic semigroup of a non-returning language has $(n-1)^n$ elements and requires at least $\binom{n}{2}$ generators. We find the maximal state complexities of atoms of non-returning languages. Finally, we show that there exists a most complex non-returning language that meets the bounds for all these complexity measures.

Keywords: atom, boolean operation, concatenation, different alphabets, most complex, reversal, regular language, star, state complexity, syntactic semigroup, transition semigroup, unrestricted complexity

1 Introduction

Formal definitions are postponed until Section 2; we assume the reader is familiar with basic properties of regular languages and finite automata as described in [13,15], for example.

A deterministic finite automaton (DFA) is *non-returning* if there are no transitions into its initial state. A regular language is non-returning if its minimal DFA has that property. The *state complexity* of a regular language L is the

number of states in the minimal DFA accepting L. The state complexity of an *operation* on regular languages is defined as the maximal state complexity of the result of the operation, expressed as a function of the state complexities of the operands.

The state complexity of common operations (union, intersection, difference, symmetric difference, Kleene star, reverse and product/concatenation) was studied by Eom, Han and Jirásková [9]. They pointed out that several subclasses of regular languages have the non-returning property; these subclasses include the class of suffix-free languages (suffix codes) and its subclasses (for example, bifix-free languages), and finite languages.

A regular language $L_n(a, b, c)$ of state complexity n is defined for all $n \geqslant 3$ in Figure 1. It was shown in [2] that the sequence $(L_3(a, b, c), \ldots, L_n(a, b, c), \ldots)$ of these languages meets the upper bounds (for regular languages) on the complexities of all the basic operations on regular languages as follows: If $L(b, a)$ is $L(a, b)$ with the roles of a and b interchanged, then $L_m(a, b) \circ L_n(b, a)$ meets the bound mn for all binary boolean operations $\circ$ that depend on both arguments; if $m \neq n$, $L_m(a, b) \circ L_n(a, b)$ meets the bound mn; $(L_n(a, b))^*$ meets the bound $2^{n-1} + 2^{n-2}$; $(L_n(a, b, c))^R$ meets the bound 2^n for reversal; and $L_m(a, b, c)L_n(a, b, c)$ meets the bound $(m-1)2^n + 2^{n-1}$ for product.
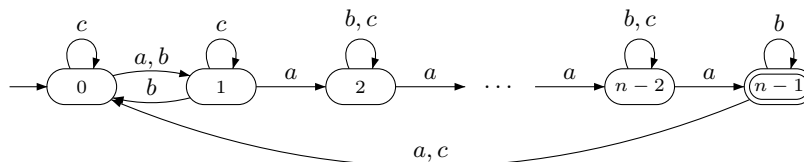

Fig. 1. Most complex regular language $L_n(a, b, c)$.

It was proposed in [2] that the size of the *syntactic semigroup* of a regular language is another worthwhile measure of the complexity of the language. The syntactic semigroup is isomorphic to the *transition semigroup* of the minimal DFA of L, that is, the semigroup of transformations of the state set of the DFA induced by non-empty words.

Another complexity measure suggested in [2] is the number and state complexities of the atoms of the language. An atom is a certain kind of intersection of complemented and uncomplemented quotients of L, where a quotient of $L \subseteq \Sigma^*$ by a word $w \in \Sigma^*$ is the language $w^{-1}L = \{x \mid wx \in L\}$.

It was shown in [2] that the languages $L_n(a, b, c)$ not only meet the bounds on the state complexities of operations, but also have the largest syntactic semigroups (of size n^n), and the largest number of atoms (2^n), all of which have the maximal possible state complexities. In this sense these are *most complex* regular languages.

In this paper we show that there also exist most complex non-returning languages. For each $n \geqslant 4$, we define a language of state complexity n. We prove

that the syntactic semigroup of this language has $(n-1)^n$ elements, that it is generated by $\binom{n}{2}$ elements, and that the number of generators cannot be reduced. We also show that this language has 2^n atoms, all of which have maximal state complexity. We demonstrate that the upper bound on the state complexity of reversal is met by a single ternary language, and that no binary language meets this bound. Moreover, restrictions of this language to binary alphabets meet the bounds for star, product and boolean operations. This is in contrast to [9] where several types of witnesses are used to meet the various bounds. Thus we correct an error in [9, Table 1], where it is stated that the upper bound on the complexity of product cannot be reached with binary witnesses. Finally, we also derive upper bounds on *unrestricted* binary operations [3], that is, operations on languages over different alphabets, and show that our non-returning language meets these bounds.

2   Preliminaries

A *deterministic finite automaton (DFA)* is a quintuple $\mathcal{D} = (Q, \Sigma, \delta, q_0, F)$, where Q is a finite non-empty set of *states*, Σ is a finite non-empty *alphabet*, $\delta \colon Q \times \Sigma \to Q$ is the *transition function*, $q_0 \in Q$ is the *initial* state, and $F \subseteq Q$ is the set of *final* states. We extend δ to a function $\delta \colon Q \times \Sigma^* \to Q$ as usual. A DFA $\mathcal{D}$ *accepts* a word $w \in \Sigma^*$ if $\delta(q_0, w) \in F$. The language accepted by $\mathcal{D}$ is denoted by $L(\mathcal{D})$. If q is a state of $\mathcal{D}$, then the language L^q of q is the language accepted by the DFA $(Q, \Sigma, \delta, q, F)$. A state is *empty* if its language is empty. Two states p and q of $\mathcal{D}$ are *equivalent* if $L^p = L^q$. A state q is *reachable* if there exists $w \in \Sigma^*$ such that $\delta(q_0, w) = q$. A DFA is *minimal* if all of its states are reachable and no two states are equivalent.

A *nondeterministic finite automaton (NFA)* is a quintuple $\mathcal{D} = (Q, \Sigma, \delta, I, F)$ where Q, Σ and F are defined as in a DFA, $\delta \colon Q \times \Sigma \to 2^Q$ is the *transition function*, and $I \subseteq Q$ is the *set of initial states*.

We use $Q_n = \{0, \ldots, n-1\}$ as our basic set with n elements. A *transformation* of Q_n is a mapping $t \colon Q_n \to Q_n$. The *image* of $q \in Q_n$ under t is denoted by qt, and this notation is extended to subsets of Q_n. The *rank* of a transformation t is the cardinality of $Q_n t$. If s and t are transformations of Q_n, their composition is denoted $(qs)t$ when applied to $q \in Q_n$. Let $\mathcal{T}_{Q_n}$ be the set of all n^n transformations of Q_n; then $\mathcal{T}_{Q_n}$ is a monoid under composition.

For $k \geqslant 2$, a transformation t of a set $P = \{q_0, q_1, \ldots, q_{k-1}\} \subseteq Q_n$ is a *k-cycle* if $q_0 t = q_1, q_1 t = q_2, \ldots, q_{k-2}t = q_{k-1}, q_{k-1}t = q_0$. This k-cycle is denoted by $(q_0, q_1, \ldots, q_{k-1})$, and leaves the states in $Q_n \setminus P$ unchanged. A 2-cycle (q_0, q_1) is called a *transposition*. A transformation that sends state p to q and acts as the identity on the remaining states is denoted by $(p \to q)$. If a transformation of Q_n has rank $n-1$, then there is exactly one pair of distinct elements $i, j \in Q_n$ such that $it = jt$. We say a transformation t of Q_n is of *type* $\{i, j\}$ if t has rank $n-1$ and $it = jt$ for $i < j$.

The *syntactic congruence* of a language $L \subseteq \Sigma^*$ is defined on Σ^+ as follows: For $x, y \in \Sigma^+, x \approx_L y$ if and only if $wxz \in L \Leftrightarrow wyz \in L$ for all $w, z \in \Sigma^*$. The

quotient set $\Sigma^+/\approx_L$ of equivalence classes of $\approx_L$ is a semigroup, the *syntactic semigroup* T_L of L.

Let $\mathcal{D} = (Q_n, \Sigma, \delta, 0, F)$ be a DFA. For each word $w \in \Sigma^*$, the transition function induces a transformation δ_w of Q_n by w: for all $q \in Q_n$, $q\delta_w = \delta(q, w)$. The set $T_{\mathcal{D}}$ of all such transformations by non-empty words is the *transition semigroup* of $\mathcal{D}$ under composition [14]. Sometimes we use the word w to denote the transformation it induces; thus we write qw instead of $q\delta_w$. We extend the notation to sets: if $P \subseteq Q_n$, then $Pw = \{pw \mid p \in P\}$. We also find it convenient to write $P \xrightarrow{w} Pw$ to indicate that the image of P under w is Pw.

If $\mathcal{D}$ is a minimal DFA of L, then $T_{\mathcal{D}}$ is isomorphic to the syntactic semigroup T_L of L [14], and we represent elements of T_L by transformations in $T_{\mathcal{D}}$. The size of this semigroup has been used as a measure of complexity [2,8,10,12].

Atoms are defined by a left congruence, where two words x and y are equivalent whenever $ux \in L$ if and only if $uy \in L$ for all $u \in \Sigma^*$. Thus x and y are equivalent whenever $x \in u^{-1}L$ if and only if $y \in u^{-1}L$ for all $u \in \Sigma^*$. An equivalence class of this relation is an *atom* of L [7]. Thus an atom is a non-empty intersection of complemented and uncomplemented quotients of L. The number of atoms and their state complexities were suggested as possible measures of complexity of regular languages [2], because all the quotients of a language, and also the quotients of atoms, are always unions of atoms [6,7,11].

Suppose $\circ$ is a unary operation on languages, and $f(n)$ is an upper bound on the state complexity of this operation. If the state complexity of $(L_n)^\circ$ is $f(n)$, then L_n is called a *witness* to the state complexity of $\circ$ for that n. In general, we need a sequence $(L_k, L_{k+1}, \ldots,)$ of such languages; this sequence is called a *stream*. Often a stream does not start at 1 because the bound may not hold for small values of n. For a binary operation we need two streams. The languages in a stream usually have the same form and differ only in the parameter n.

Sometimes the same stream can be used for both operands of a binary operation, but this is not always possible. For example, for boolean operations when $m = n$, the state complexity of $L_n \cup L_n = L_n$ is n, whereas it should be $mn = n^2$. However, in many cases the second language is a "dialect" of the first, that is, it "differs only slightly" from the first. A *dialect* of $L_n(\Sigma)$ is a language obtained from $L_n(\Sigma)$ by deleting some letters of Σ in the words of $L_n(\Sigma)$ – by this we mean that words containing these letters are deleted – or replacing them by letters of another alphabet Σ'. In this paper we will encounter only two types of dialects:

1. A dialect in which some letters were deleted; for example, $L_n(a, b)$ is a dialect of $L_n(a, b, c)$ with c deleted, and $L_n(a, -, c)$ is a dialect with b deleted.
2. A dialect in which the roles of two letters are exchanged; for example, $L_n(b, a)$ is such a dialect of $L_n(a, b)$.

These two types of dialects can be combined, for example, in $L_n(a, -, b)$ the letter c is deleted, and b plays the role that c played originally. The notion of dialects also extends to DFAs; for example, if $\mathcal{D}_n(a, b, c)$ recognizes $L_n(a, b, c)$ then $\mathcal{D}_n(a, -, b)$ recognizes the dialect $L_n(a, -, b)$.

3 Main Results

From now on by *complexity* we mean *state complexity*. If a word w induces a transformation t in a DFA, we denote this by $w : t$.

Let $\Gamma = \{a_{i,j} \mid 0 \leqslant i < j \leqslant n-1\}$, where $a_{i,j}$ is a letter that induces any transformation of type $\{i, j\}$ and does not map any state to 0. Let $\Gamma' = \Gamma \setminus \{a_{0,n-1}, a_{0,1}, a_{1,n-1}, a_{1,2}\}$. Let $\Sigma = \{a, b, c, d\} \cup \Gamma'$, where $a : (1, \ldots, n-1)(0 \to 1)$, $b : (1, 2)(0 \to 2)$, $c : (2, \ldots, n-1)(1 \to 2)(0 \to 1)$, and $d : (0 \to 2)$. That is:

- $qa = q + 1$ for $0 \leqslant q \leqslant n-2$, and $(n-1)a = 1$.
- $0b = 2$, $1b = 2$, $2b = 1$ and $qb = q$ for $q \notin \{0, 1, 2\}$.
- $qc = q + 1$ for $0 \leqslant q \leqslant n-2$, and $(n-1)c = 2$.
- $0d = 2$ and $qd = q$ for $q \neq 0$.

Note that a, b, c and d are transformations of types $\{0, n-1\}$, $\{0, 1\}$, $\{1, n-1\}$ and $\{0, 2\}$, respectively. Note also that a, b and c restricted to $Q_n \setminus \{0\}$ generate all the transformations of $\{1, \ldots, n-1\}$. This follows from the well-known fact that the full transformation semigroup on a set X can be generated by the symmetric group on X together with a transformation of X with rank $|X| - 1$. For $X = \{1, \ldots, n-1\}$, we see that $\{(1, \ldots, n-1), (1, 2)\}$ (the restrictions of a and b) generate the symmetric group, and $(2, \ldots, n-1)(1 \to 2)$ (the restriction of c) is a transformation of rank $|X| - 1 = n - 2$.

We are now ready to define a most complex non-returning DFA and language.

Definition 1. *For $n \geqslant 4$, let $\mathcal{D}_n = \mathcal{D}_n(\Sigma) = (Q_n, \Sigma, \delta_n, 0, \{n-1\})$, where $\Sigma = \{a, b, c, d\} \cup \Gamma'$, and δ_n is defined in accordance with the transformations described above. See Figure 2. Let $L_n = L_n(\Sigma)$ be the language accepted by $\mathcal{D}_n(\Sigma)$.*
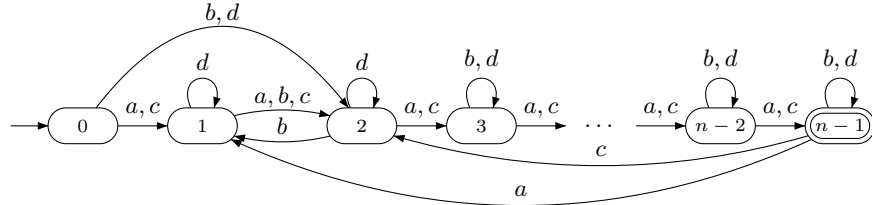

Fig. 2. Most complex non-returning language $L_n(\Sigma)$ of Definition 1. The letters in $\Gamma' = \Sigma \setminus \{a, b, c, d\}$ are omitted.

Theorem 1 (Most Complex Non-Returning Languages). *For each $n \geqslant 4$, the DFA of Definition 1 is minimal and non-returning. The stream $(L_n(\Sigma) \mid n \geqslant 4)$ with some dialect streams is most complex in the class of regular non-returning languages in the following sense:*

1. *The syntactic semigroup of $L_n(\Sigma)$ has cardinality $(n-1)^n$, and at least $\binom{n}{2}$ letters are required to reach this bound.*

2. *Each quotient of $L_n(a)$ has complexity $n-1$, except L itself, which has complexity n.*
3. *The reverse of $L_n(a,b,c)$ has complexity 2^n, and at least three letters are needed to meet this bound. Moreover, $L_n(a,b,c)$ has 2^n atoms.*
4. *For each atom A_S of $L_n(\Sigma)$, the complexity $\kappa(A_S)$ satisfies:*

$$\kappa(A_S) \leqslant \begin{cases} 2^{n-1}, & \text{if } S \in \{\emptyset, Q_n\}; \\ 2 + \sum_{x=1}^{|S|} \sum_{y=1}^{|S|} \binom{n-1}{x}\binom{n-1-x}{y}, & \text{if } \emptyset \subsetneq S \subsetneq Q_n. \end{cases}$$

   Moreover, at least $\binom{n}{2}$ letters are required to meet these bounds.
5. *The star of $L_n(a,b)$ has complexity 2^{n-1}.*
6. (a) *Restricted product: $\kappa(L_m(a,b)L_n(a,-,b)) = (m-1)2^{n-1} + 1$.*
   (b) *Unrestricted product: For $m, n \geqslant 4$, let L_m (respectively, L_n) be a non-returning language of complexity m (respectively, n) over an alphabet Σ', (respectively, Σ). Then the complexity of product is at most $m2^{n-1} + 1$, and this bound is met by $L_m(a,b)$ and $L_n(a,-,b,d)$.*
7. (a) *Restricted boolean operations: $\kappa(L_m(a,b) \circ L_n(b,a)) = mn - (m+n-2)$.*
   (b) *In case $m \neq n$, $\kappa(L_m(a,b) \circ L_n(a,b)) = mn - (m+n-2)$.*
   (c) *Unrestricted boolean operations: The complexity of $L_m(a,b,c) \circ L_n(b,a,d)$ is $mn+1$ if $\circ \in \{\cup, \oplus\}$, that of $L_m(a,b,c) \setminus L_n(b,a)$ is $mn-n+2$, and of $L_m(a,b) \cap L_n(b,a)$ is $mn - (m+n-2)$.*

All of these bounds are maximal for non-returning languages.

Proof. From the definition of the letters of Σ it is obvious that the DFA $\mathcal{D}_n$ is non-returning, and that any pair (p,q) of states can be distinguished by the shortest word in a^* accepted by p but not by q.

1. This follows from Propositions 1 and 2.
2. Obvious.
3. By Proposition 3 the number of atoms of $L_n(a,b,c)$ is 2^n. By [7] the complexity of the reverse is the same as the number of atoms. By Proposition 4 at least three letters are required to meet this bound on the number of atoms and the complexity of reverse.
4. See Propositions 5, 6, and 7.
5. See Proposition 8.
6. See Propositions 9 and 10.
7. See Propositions 11 and 12.

The rest of this paper is devoted to the proofs of the propositions above. □

4   Syntactic Semigroup

For all basic operations on non-returning languages, the complexity bounds can be met with either binary or ternary witnesses [9]. However, to meet the bound for the size of the syntactic semigroup, our most complex stream is forced to use an alphabet that grows quadratically in size.

Let $Q_n = \{0, 1, \ldots, n-1\}$. For $n \geqslant 2$, let N_n denote the semigroup of transformations of Q_n such that $it \neq 0$ for all $i \in Q_n$. We call N_n the *full non-returning semigroup* of degree n. We will show that a minimal generating set for N_n must have size $\binom{n}{2}$.

Proposition 1. *If G is a generating set for N_n, then G contains a transformation of type $\{i, j\}$ for each set $\{i, j\} \subseteq Q_n$. In particular, a minimal generating set has exactly one element of type $\{i, j\}$ for each $\{i, j\} \subseteq Q_n$, and there are $\binom{n}{2}$ such elements.*

Proof. Suppose t is a transformation of type $\{i, j\}$, and let t' be an arbitrary transformation. If tt' has rank $n-1$, then tt' has type $\{i, j\}$. Indeed, since $it = jt$, it follows that $itt' = jtt'$. Thus composing a transformation of type $\{i, j\}$ with an arbitrary transformation either preserves the type, or lowers the rank.

Now, suppose G generates N_n. Observe that N_n does not contain transformations of rank n, since such transformations necessarily map some element to 0. Since composition with a transformation of type $\{i, j\}$ either preserves type or lowers rank, the semigroup generated by G contains only transformations that either have the same type as some element of G, or have rank less than $n-1$ and so are typeless. But N_n contains a transformation of type $\{i, j\}$ for each $\{i, j\} \subseteq Q_n$. So if G generates N_n, then G must contain an element of type $\{i, j\}$ for each $\{i, j\} \subseteq Q_n$. $\qquad\square$

This gives a necessary condition for G to generate N_n. Now we give a sufficient condition.

Proposition 2. *Let G be a subset of N_n that contains a transformation of type $\{i, j\}$ for each set $\{i, j\} \subseteq Q_n$. Let G' be obtained by restricting every transformation in G to $Q_n \setminus \{0\}$. If G' generates the symmetric group on $Q_n \setminus \{0\}$, then G generates N_n.*

Proof. First, we show that G' in fact generates the full transformation semigroup on $Q_n \setminus \{0\}$. Recall that the full transformation semigroup on X is generated by the symmetric group on X together with a transformation of X of rank $|X| - 1$. By assumption, G' contains generators of the symmetric group on $Q_n \setminus \{0\}$. Transformations in G of type $\{i, j\}$ with $0 < i, j$ have rank $n-1$, and furthermore their restrictions to $Q_n \setminus \{0\}$ are of rank $n-2$. Thus G' contains generators of the symmetric group on $Q_n \setminus \{0\}$, as well as a transformation of rank $|Q_n \setminus \{0\}| - 1 = n-2$; it follows that G' generates the full transformation semigroup on $Q_n \setminus \{0\}$.

Now, we prove that G generates every transformation in N_n. Let t be an element of N_n; we want to show that t is in the semigroup generated by G. Since N_n does not contain any transformations of rank n, the transformation t has rank less than n, and thus there exist distinct $i, j \in Q_n$ such that $it = jt$. Select a transformation s of type $\{i, j\}$ in G. Then for distinct $q, q' \in Q_n$, we have $qs = q's$ if and only if $\{q, q'\} = \{i, j\}$.

Hence there is a well-defined transformation r' of $Q_n \setminus \{0\}$ given by $(qs)r' = qt$ for all $q \in Q$; it is well-defined since if we have $qs = q's$, then $\{q, q'\} = \{i, j\}$ and

is and *js* get mapped to a common element $it = jt$. The transformation r' lies in the full transformation semigroup on $Q_n \setminus \{0\}$, and so it is in the semigroup generated by G'. Hence there is some transformation r of Q_n in the semigroup generated by G such that r is equal to r' when restricted to $Q_n \setminus \{0\}$.

Since $qs \in Q_n \setminus \{0\}$ for all $q \in Q_n$, it follows that $(qs)r = (qs)r' = qt$ for all $q \in Q_n$, and thus sr and t are equal as transformations. Since s is in G and r is in the semigroup generated by G, it follows $sr = t$ is in the semigroup generated by G. Thus the semigroup generated by G contains all elements of N_n; but G is a subset of N_n, so G generates N_n.   □

5   Number and Complexities of Atoms

Denote the complement of a language L by $\overline{L} = \Sigma^* \setminus L$. Let $Q_n = \{0, \ldots, n-1\}$ and let L_n be a non-empty regular language with quotients $K = \{K_0, \ldots, K_{n-1}\}$. Each subset S of Q_n defines an *atomic intersection* $A_S = \bigcap_{i \in S} K_i \cap \bigcap_{i \in \overline{S}} \overline{K_i}$, where $\overline{S} = Q_n \setminus S$. An *atom* of L is a non-empty atomic intersection; this definition is equivalent to that given in Section 2 in terms of a left congruence. Note that if $S \neq T$, then $A_S \cap A_T = \emptyset$; hence atoms corresponding to distinct subsets of Q_n are disjoint. A language of complexity n can have at most 2^n atoms, since there are 2^n subsets of Q. We show that this bound can be met by non-returning languages. Additionally, we derive upper bounds on the complexities of atoms of non-returning languages, and show that our most complex stream meets these bounds.

We now describe a construction due to Iván [11]. Let L be a regular language with DFA $\mathcal{D} = (Q, \Sigma, \delta, q_0, F)$. For each $S \subseteq Q$, we define a DFA $\mathcal{D}_S = (Q_S, \Sigma, \Delta, (S, \overline{S}), F_S)$ as follows.

- $Q_S = \{(X, Y) : X, Y \subseteq Q, X \cap Y = \emptyset\} \cup \{\bot\}$. State $\bot$ is the *sink state*.
- $\Delta((X, Y), a) = (Xa, Ya)$ if $Xa \cap Ya = \emptyset$, and otherwise $\Delta((X, Y), a) = \bot$; also $\Delta(\bot, a) = \bot$.
- $F_S = \{(X, Y) : X \subseteq F, Y \subseteq \overline{F}\}$.

The DFA $\mathcal{D}_S$ recognizes the atomic intersection A_S of L; if it recognizes a non-empty language, then A_S is an atom. We can determine the complexity of A_S by counting reachable and distinguishable states in $\mathcal{D}_S$.

5.1   Number of Atoms

Proposition 3. *The language $L_n = L_n(a, b, c)$ has 2^n atoms.*

Proof. We want to show that A_S is an atom of L_n for all $S \subseteq Q_n$. It suffices to show for each S that the DFA $\mathcal{D}_S$ recognizes at least one word. Then since atoms corresponding to different subsets of Q_n are disjoint, this proves there are 2^n distinct atoms.

First, we show that from the initial state $(S, \overline{S})$, we can reach some state of the form (X, Y) where $0 \notin X$ and $0 \notin Y$. Consider the set $\{0, 1, n-1\}$.

Notice that for each subset $\{i,j\}$ of $\{0, 1, n-1\}$, we have a transformation of type $\{i,j\}$: a has type $\{0, n-1\}$, b has type $\{0,1\}$, and c has type $\{1, n-1\}$. Additionally, by the pigeonhole principle, either S contains two distinct elements from $\{0, 1, n-1\}$, or $\overline{S}$ contains two distinct elements from $\{0, 1, n-1\}$.

Suppose without loss of generality it is S which contains two distinct elements from $\{0, 1, n-1\}$. Let $\{i,j\} \subseteq S$ for some $\{i,j\} \subseteq \{0, 1, n-1\}$ with $i \neq j$. Let $\sigma \in \Sigma$ be the letter inducing the transformation of type $\{i,j\}$. Then we claim $(S, \overline{S})\sigma \neq \perp$. Indeed, suppose that $q \in S\sigma \cap \overline{S}\sigma$. Then since σ is a transformation of type $\{i,j\}$, we must have $i\sigma = j\sigma = q$, and no other element is mapped to q. But $\{i,j\} \subseteq S$, so we cannot have $q \in \overline{S}\sigma$.

Hence $S\sigma \cap \overline{S}\sigma = \emptyset$. Furthermore, since σ is a non-returning transformation, we have $0 \notin S\sigma$ and $0 \notin \overline{S}\sigma$. Thus starting from the initial state $(S, \overline{S})$, we can apply σ to reach a state of the form (X, Y) with $0 \notin X$ and $0 \notin Y$.

Now, recall that the three transformations $\{a, b, c\}$, when restricted to $Q_n \setminus \{0\}$, generate all transformations of $Q_n \setminus \{0\}$. Since $X \subseteq Q_n \setminus \{0\}$, there exists a transformation of $Q_n \setminus \{0\}$ that maps every element of X to $n-1$ and every element of $(Q_n \setminus \{0\}) \setminus X$ to 1. Let $w \in \{a, b, c\}^*$ be a word that induces this transformation when restricted to $Q_n \setminus \{0\}$. Since $Y \subseteq Q_n \setminus \{0\}$ and Y is disjoint from X, it follows that w maps every element of Y to 1. Since $F_n = \{n-1\}$ is the final state set of $\mathcal{D}_n$, we see that $Xw \subseteq F_n$ and $Yw \subseteq \overline{F_n}$. Thus $(Xw, Yw) = (\{n-1\}, \{1\})$ is a final state of $\mathcal{D}_S$.

This shows that there exists a word $\sigma w \in \{a, b, c\}^*$ that maps the initial state $(S, \overline{S})$ of $\mathcal{D}_S$ to a final state. Thus A_S is an atom. □

Next, we prove that the bound on number of atoms cannot be met by a binary witness. From [7] we know that the number of atoms of a regular language is equal to the state complexity of the reversal of the language. Hence this also proves a conjecture from [9], that a ternary witness is necessary to meet the bound for reversal of non-returning languages.

Proposition 4. *Let L be a non-returning language of complexity n over $\Sigma = \{a, b\}$. Then the number of atoms of L is strictly less than 2^n.*

Proof. Let $\mathcal{D}$ be the minimal DFA of L, with state set Q_n. We introduce some special terminology for this proof, which generalizes the notion of transformations of type $\{i,j\}$. We say that a transformation t *unifies* i and j, or unifies the set $\{i,j\}$, if $it = jt$. For example, transformations of type $\{i,j\}$ unify $\{i,j\}$. But furthermore, every transformation of Q_n of rank $n-1$ or less unifies at least one pair of elements of Q_n. The transition semigroup of $\mathcal{D}$ cannot have transformations of rank n, since L is non-returning; thus all the transformations in the transition semigroup must unify some pair of states.

Suppose that in $\mathcal{D}$, the letter a induces a transformation that unifies $\{i,j\}$, and b induces a transformation that unifies $\{k, \ell\}$. Assume also that $i \neq j$ and $k \neq \ell$. We will show that at least one atomic intersection A_S of L is empty, and thus is not an atom.

Suppose $\{i,j\} = \{k, \ell\}$. Let $S = \{i\}$ and consider the atomic intersection A_S. The initial state of the DFA for A_S is $(\{i\}, \overline{S})$. Note that $j \in \overline{S}$, so $ja \in \overline{S}a$.

But a unifies i and j, so $ja = ia \in \{i\}a$. Thus since $\{i\}a \cap \overline{S}a \neq \emptyset$, the letter a sends the initial state $(\{i\}, \overline{S})$ to the sink state. Since b also unifies i and j, the letter b also sends $(\{i\}, \overline{S})$ to the sink state. Thus A_S is non-empty if and only if $(\{i\}, \overline{S})$ is a final state. In fact, either A_S is non-empty or $A_S = \{\varepsilon\}$, since every non-empty word sends the initial state $(\{i\}, \overline{S})$ to the sink state. If we let $T = \{j\}$, the same argument shows that A_T is either empty or $A_T = \{\varepsilon\}$. But $A_S \cap A_T = \emptyset$, so one of A_S or A_T must be empty.

Now, suppose $\{i, j\} \cap \{k, \ell\} = \emptyset$. Let $S = \{i, k\}$ and consider the atomic intersection A_S. The initial state of the DFA for A_S is $(\{i, k\}, \overline{S})$ with $j, \ell \in \overline{S}$. Thus similarly to before, the transformation a which unifies $\{i, j\}$ and the transformation b which unifies $\{k, \ell\}$ both send A_S to a sink state. So either A_S is empty or $A_S = \{\varepsilon\}$. For $T = \{j, \ell\}$, the same argument shows that either A_T is empty or $A_T = \{\varepsilon\}$. Hence as before, one of A_S or A_T is empty.

Finally, suppose $\{i, j\} \cap \{k, \ell\}$ has exactly one element. Then either $k \in \{i, j\}$ or $\ell \in \{i, j\}$. Assume without loss of generality that $\ell \in \{i, j\}$ and $\ell = i$; otherwise rename the elements so this is the case. Then a unifies $\{i, j\}$, and b unifies $\{i, k\}$. Let $S = \{i\}$ and consider A_S. As before, the initial state of the DFA for A_S is sent to a sink state by both a and b. Thus either A_S is empty or $A_S = \{\varepsilon\}$. For $T = \{j, k\}$, the same argument shows that either A_T is empty or $A_T = \{\varepsilon\}$. Hence one of A_S or A_T is empty. $\square$

5.2   Complexity of Atoms

First, we derive upper bounds for the complexity of atoms of non-returning languages.

Proposition 5. *Let L be a non-returning language of complexity n, and let Q_n be the state set of its minimal DFA. Let $S \subseteq Q_n$; then we have*

$$\kappa(A_S) \leqslant \begin{cases} 2^{n-1}, & \text{if } S \in \{\emptyset, Q_n\}; \\ 2 + \sum_{x=1}^{|S|} \sum_{y=1}^{|S|} \binom{n-1}{x}\binom{n-1-x}{y}, & \text{if } \emptyset \subsetneq S \subsetneq Q_n. \end{cases}$$

Proof. To obtain an upper bound on the complexity of the atomic intersection A_S, we find an upper bound on the number of reachable states in the DFA $\mathcal{D}_S$.

Let $S = \emptyset$; then the initial state of $\mathcal{D}_S$ is $(\emptyset, Q_n)$. From this state, we can only reach states of the form $(\emptyset, X)$ with $X \subseteq Q_n \setminus \{0\}$ and X non-empty. The fact that $0 \notin X$ follows from the non-returning property. There are $2^{n-1} - 1$ non-empty subsets of $Q_n \setminus \{0\}$; adding 1 for the initial state gives the upper bound of 2^{n-1}. For $S = Q_n$ we use a symmetric argument. Note that if $S = \emptyset$ or $S = Q_n$, the sink state $\bot$ is not reachable in $\mathcal{D}_S$.

Now suppose $S \neq \emptyset$ and $S \neq Q_n$. The initial state of $\mathcal{D}_S$ is $(S, \overline{S})$. By the non-returning property, this is the only state (X, Y) of $\mathcal{D}_S$ that has $0 \in X$ or $0 \in Y$. Hence all non-initial, non-sink reachable states have the form (X, Y) where $1 \leqslant |X| \leqslant |S|$, $1 \leqslant |Y| \leqslant |\overline{S}| = n - |S|$, $X \cap Y = \emptyset$, $0 \notin X$, and $0 \notin Y$.

Suppose $|X| = x$. There are $\binom{n-1}{x}$ subsets of $Q_n \setminus \{0\}$ of size x. Given a subset X, if we want to choose a set $Y \subseteq Q_n \setminus \{0\}$ which has size y and is

disjoint from X, there are $\binom{n-1-x}{y}$ choices: we must choose y elements from $Q_n \setminus \{0\}$, but we cannot choose any of the x elements from X, so there are really $n - 1 - x$ options to choose from. Hence the number of non-initial, non-sink reachable states (X, Y) with $|X| = x$ and $|Y| = y$ is bounded by

$$\binom{n-1}{x}\binom{n-1-x}{y}.$$

Since $|X|$ can range from 1 to $|S|$ and $|Y|$ can range from 1 to $n - |S|$, we take the sum over thse ranges to get a bound on the total number of non-initial, non-sink reachable states:

$$\sum_{x=1}^{|S|} \sum_{y=1}^{n-|S|} \binom{n-1}{x}\binom{n-1-x}{y}.$$

Finally, add 2 for the initial and sink states. This gives the stated bound.   □

Next, we prove our witness meets these bounds. First, we recall a lemma from [4].

Lemma 1. *Let L be a regular language, and let $\mathcal{D}_S$ be the DFA of the atomic intersection A_S of L. Then:*

1. *States (X, Y) and (X', Y') of $\mathcal{D}_S$ are distinguishable if $X \neq X'$ and $A_X, A_{X'}$ are both atoms of L, or if $Y \neq Y'$ and $A_{\overline{Y}}, A_{\overline{Y'}}$ are both atoms of L.*
2. *If one of A_X or $A_{\overline{Y}}$ is an atom of L, then (X, Y) is distinguishable from $\bot$.*

Proposition 6. *The atoms of the language $L_n = L_n(\Sigma)$ meet the complexity bounds of Proposition 5.*

Proof. By Proposition 3, A_S is an atom of L_n for each $S \subseteq Q_n$. Thus by Lemma 1, for each $S \subseteq Q_n$, all distinct states of $\mathcal{D}_S$ are distinguishable. So it suffices to just show that the number of reachable states of each atom A_S meets the bound. Recall that the transformations in Σ generate the full non-returning semigroup N_n, which contains all transformations of Q_n that do not map anything to 0.

Suppose $S = \emptyset$. The initial state of $\mathcal{D}_S$ is $(\emptyset, Q_n)$. We claim that $(\emptyset, X)$ is reachable for all non-empty $X \subseteq Q_n \setminus \{0\}$. Indeed, just let $w \in \Sigma^*$ induce a transformation that maps Q_n onto X; then $(\emptyset, Q_n)w = (\emptyset, X)$. Hence there are 2^{n-1} reachable states. A symmetric argument works for $S = Q_n$.

Suppose $S \neq \emptyset$ and $S \neq Q_n$. We see from the proof of Proposition 5 that to show that the number of reachable states of $\mathcal{D}_S$ meets the bound, it suffices to show that every state (X, Y) with $1 \leqslant |X| \leqslant |S|$, $1 \leqslant |Y| \leqslant |\overline{S}|$, $X \cap Y$, $0 \notin X$, and $0 \notin Y$ is reachable, in addition to the sink state $\bot$. The sink state can be reached from the initial state $(S, \overline{S})$ using a constant transformation. To reach (X, Y), simply use a transformation that maps S onto X and $\overline{S}$ onto Y; this is possible since X and Y are disjoint, and also since $X, Y \subseteq Q_n \setminus \{0\}$ and we have all transformations of Q_n that do not map anything to 0. Hence the number of reachable states matches the bound.   □

We used the full alphabet of our witness in the previous proof. The following proposition shows that this was necessary.

Proposition 7. *Let L be a non-returning language over Σ of complexity n. If the atoms of L meet the bounds of Proposition 5, then Σ has size at least $\binom{n}{2}$.*

Proof. We claim that to meet the bounds, Σ must contain a letter inducing a transformation of type $\{i, j\}$ for each $\{i, j\} \subseteq Q_n$. There are $\binom{n}{2}$ of these subsets.

To see this, fix $\{i, j\}$ and suppose no letter induces a transformation of type $\{i, j\}$. Then the transition semigroup of the minimal DFA of L does not contain a transformation of type $\{i, j\}$, since there is no transformation of this type in its generating set.

Now, let $S = \{i, j\}$ and consider the atomic intersection A_S. The initial state of the DFA for this atomic intersection is $(\{i, j\}, \overline{\{i, j\}})$. From this state, there is no way to reach a state of the form (X, Y) with $|X| = 1$ and $|Y| = n - 2$, since this would require a transformation of type $\{i, j\}$. Hence A_S cannot have maximal complexity. $\qquad\square$

6   Star

Proposition 8 (Star). *Let $\mathcal{D}_n(a, b)$ be the DFA of Definition 1 and let $L_n(a, b)$ be its language. Then the complexity of $(L_n(a, b))^*$ is 2^{n-1}.*

Proof. The upper bound of 2^{n-1} on the complexity of star was established in [9]. We use the construction of [9] for an NFA for $(L_n(a, b))^*$ as shown in Figure 3: state 0 has been made final and a b-transition has been added from state $n - 1$ to state 2.
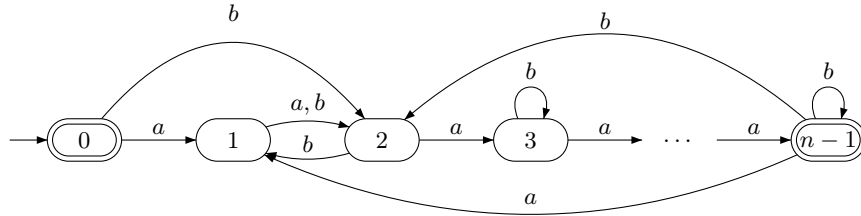

Fig. 3. NFA for star of $L_n(a, b)$.

State $\{0\}$ is initial and for $q \in \{1, \ldots, n - 1\}$, the set $\{q\}$ is reachable by a^q.

We prove by induction on the size of the set that all nonempty subsets of $\{1, \ldots, n - 1\}$ are reachable. We can think about the elements of the set $S = \{q_1, \ldots, q_k\}$, $1 \leqslant q_1 < \cdots < q_k \leqslant n - 1$, as a pattern of states that are in the set. Applying a adds 1 to each state or moves the pattern one position to the right

cyclically. Applying a^{n-2} subtracts 1 from each state, or moves the pattern one position to the left cyclically.

1. $q_k < n - 1$. To reach set $S = \{q_1, \ldots, q_k\}$, start with set $T = \{q_2, \ldots, q_k\}$, which is reachable by the induction assumption. Note that $q_2 \geqslant 2$. Apply a^{n-1-q_k} so that the pattern is shifted to the right until q_k reaches $n - 1$. Use b to add 2. Now we have the required k elements, but the distance between 2 and $q_2 + n - 1 - q_k$ may be larger than that between q_1 and q_2. It cannot be smaller because $n - 1 \geqslant q_k - q_1 + 2$, and so $(q_2 + (n - 1 - q_k)) - 2 \geqslant q_2 - q_1$.

   If $q_2 + (n-1-q_k) - 2 = q_2 - q_1$, then the distance between 2 and $q_2 + (n-1-q_k)$ is equal to the distance between q_1 and q_2. Thus we reach S if we shift the pattern by a word in a^* that sends 2 to q_1.

   Otherwise, we have $q_2 + (n - 1 - q_k) - 2 > q_2 - q_1$, that is, $n - 1 > q_k - q_1 + 2$. Use a^{n-1} to move the pattern, including 2, to the left one position. Since $q_2 \geqslant q_1 + 1$, we have $n - 1 > q_k - q_2 + 3$ and $q_2 + (n - 1 - q_k) - 1 > 2$. Now use b to move 1 to 2. This decreases the distance between 2 and $q_2 + (n - 1 - q_k)$ by one. This can be repeated. If we reach the pattern in which we have 2, and $q_2 + (n - 1 - q_k)$ has been moved to 3, then we are done, because the distance between q_2 and q_1 must be at least one.
2. $q_1 > 1$, $q_k = n - 1$. A shift to the left by a^{n-2} brings us to Case (1). After reaching $\{q_1 - 1, \ldots, q_k - 1\}$, the desired set can be reached by a.
3. $q_1 = 1$, $q_2 = 2, q_k = n - 1$. Use $\{q_2 = 2, q_3, \ldots, q_{k-1}, q_k = n - 1\} \xrightarrow{b} \{1, 2, q_3, \ldots, q_{k-1}, q_k = n - 1\}$.
4. $q_1 = 1, q_2 > 2, q_{k-1} = n - 2, q_k = n - 1$. Applying a we get a pattern with $\{q_1 = 1, q_2 = 2, q_3, \ldots, q_k = n - 1\}$. This is reachable by Case (3). The desired pattern is then reached by a^{n-2}.
5. $q_1 = 1, q_2 > 2, q_{k-1} < n - 2, q_k = n - 1$. We have $\{1, q_2, \ldots, n - 1\} \xrightarrow{a} \{1, 2, q_2 + 1, \ldots, q_{k-1} + 1\}$. The latter set is reachable by Case (1), and then the desired set, by a^{n-2}.

For distinguishability, $\{0\}$ is the only set with a^{n-1} as its shortest accepted non-empty word. If two subsets of $\{1, \ldots, n - 1\}$ differ by state q, then a^{n-1-q} distinguishes them. Thus our claim holds.   □

7   Product

When dealing with binary operations, to avoid confusion between the sets of states $\{0, \ldots, m - 1\}$ and $\{0, \ldots, n - 1\}$ we use $\mathcal{D}'_m(\Sigma) = (Q'_m, \Sigma, \delta'_m, 0', \{(m - 1)'\})$, and $\mathcal{D}_n(\Sigma) = (Q_n, \Sigma, \delta_n, 0, \{n - 1\})$, where $Q'_m = \{0', \ldots, (m - 1)'\}$.

Proposition 9 (Restricted Product). *Let $\mathcal{D}_n(a, b, c)$ be the DFA of Definition 1 and let $L_n(a, b, c)$ be its language. Then for $m, n \geqslant 4$ the complexity of $L'_m(a, b)L_n(a, -, b)$ is $(m - 1)2^{n-1} + 1$.*

The language $L_n(a, -, b)$ is recognized by the DFA $\mathcal{D}_n(a, -, b)$. This DFA has alphabet $\{a, b\}$ and transformations $a\colon (1, \ldots, n - 1)(0 \to 1)$ and $b\colon (2, \ldots, n -$

$1)(1 \rightarrow 2)(0 \rightarrow 1)$. That is, b induces the same transformation in $\mathcal{D}_n(a, -, b)$ that c induces in $\mathcal{D}_n(a, b, c)$.

We first establish a lemma.

Lemma 2. *Let $a: (1, \ldots, n-1)(0 \rightarrow 1)$ and $b: (2, \ldots, n-1)(1 \rightarrow 2)(0 \rightarrow 1)$. Each element of $\{1+2k, \ldots, n-1\}$ has exactly one inverse under the transformation $(ab)^k$, and the inverse of q is $q-2k$. In particular, if $S \subseteq \{1+2k, \ldots, n-1\}$, then $S(ab)^{-k} = \{q - 2k : q \in S\}$.*

Proof. We proceed by induction on k. For $k = 0$, note that $(ab)^0$ is the identity transformation. Thus each $q \in \{1, \ldots, n-1\}$ has $q - 2k = q$ as its unique inverse under $(ab)^0$.

Now, assume $k \geqslant 1$ and the result holds for all values less than k. Suppose $p(ab)^k = q$, that is, p is an inverse of q under $(ab)^k$. By the induction hypothesis, q has a unique inverse under $(ab)^{k-1}$, and this inverse is $q - 2(k-1)$. Thus $p(ab) = q(ab)^{-(k-1)} = q - 2(k-1)$. Now, since $1 + 2k \leqslant q \leqslant n-1$ and $k \geqslant 1$, we have $3 \leqslant q - 2(k-1) \leqslant n-1$. Since b is a transformation of type $\{1, n-1\}$ and $1b = (n-1)b = 2$, the only element that does not have a unique inverse under b is 2. Thus $q - 2(k-1)$ has a unique inverse under b, and this inverse is $q - 2(k-1) - 1$. We have $2 \leqslant q - 2(k-1) - 1 \leqslant n-2$. Since a is a permutation when restricted to $\{1, \ldots, n-1\}$, it follows that $q - 2(k-1) - 1$ has a unique inverse under a, and this inverse is $q - 2(k-1) - 2 = q - 2k$. Thus we must have $p = q - 2k$. $\qquad\square$

Proof (Proposition 9).

The upper bound of $(m-1)2^{n-1} + 1$ was established in [9]. The NFA construction that was used in [9] for the product is shown in Figure 4. State 0 can be omitted and its transitions replaced by appropriate transitions from the final state of $\mathcal{D}'_m$ to states 1, 2 and $n-1$. We will show that $\{0'\}$ and all subsets of the form $\{p'\} \cup S$, where $p' \in Q'_m \setminus \{0'\} = \{1', \ldots, (m-1)'\}$ and $S \subseteq Q_n \setminus \{0\} = \{1, \ldots, n-1\}$, are reachable and pairwise distinguishable.

Set $\{0'\}$ is the initial state, and $\{p'\}$ is reached by a^p. Hence our claim holds for $|S| = 0$. Next, we prove the claim for sets $\{p'\} \cup S$ with $|S| = 1$. First consider sets of the form $\{1', q\}$, $q \in Q_n \setminus \{0\}$.

– First note that $\{1'\} \xrightarrow{a^{m-1}} \{1', 1\}$.
– We have $\{1', q\} \xrightarrow{a} \{2', q+1\} \xrightarrow{b} \{1', q+2\}$ for $q + 2 \leqslant n-1$. By starting from $\{1', 1\}$ and repeatedly applying ab, we can reach all sets $\{1', q\}$ with q odd.
– If $n-1$ is odd, we can reach $\{1', n-1\}$ and then $\{1', n-1\} \xrightarrow{a} \{2', 1\} \xrightarrow{b} \{1', 2\}$. Then by repeatedly applying ab we reach all sets $\{1', q\}$ with q even, and we are done.
– If $n-1$ is even, $n-2$ is odd, so we can reach $\{1', n-2\}$. Then $\{1', n-2\} \xrightarrow{b} \{2', n-1\} \xrightarrow{b} \{1', 2\}$. By repeatedly applying ab we reach all sets $\{1', q\}$ with q even.
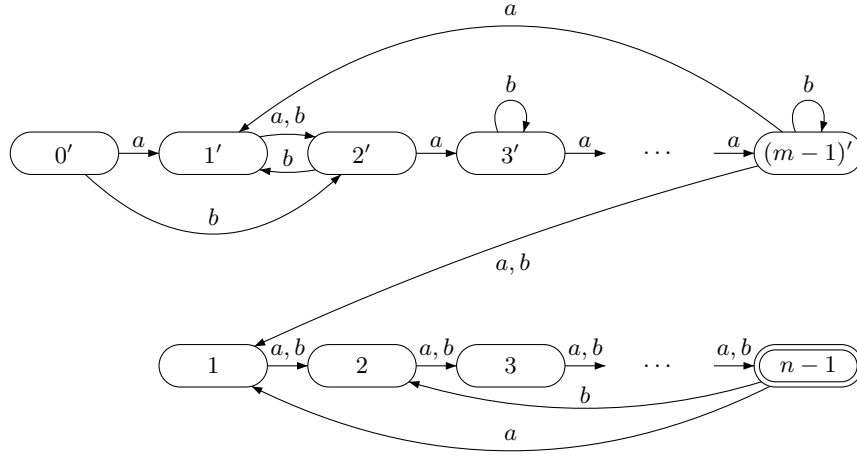
Fig. 4. NFAs for product of $L'_m(a,b)$ and $L_n(a,-,b)$.

Now, note that a restricted to $Q_n \setminus \{0\}$ is a permutation, so each $q \in Q_n \setminus \{0\}$ has an inverse under a. We can reach $\{1', qa^{-(p-1)}\}$ and then by a^{p-1} we reach $\{p', q\}$. Thus $\{p', q\}$ is reachable for each $p' \in Q'_m \setminus \{0'\}$ and $q \in Q_n \setminus \{0\}$.

We prove the claim for the remaining sets $\{p'\} \cup S$ by induction, taking the case $|S| = 1$ as our base case. Suppose $|S| > 1$, and suppose $\{p'\} \cup T$ is reachable whenever $p' \in Q'_m \setminus \{0'\}$ and $|T| < |S|$. We refer to this as the *primary induction hypothesis*, and the claim as the *primary claim*, since we will need to do another inductive proof within this proof.

First we consider the case where $p' = 1'$. Our argument for this case will be split into sub-cases based on the smallest element of S. If 1 is the smallest element of S, then by the primary induction hypothesis, we can reach $\{1'\} \cup T$ for $T = (S \setminus \{1\})a^{-(m-1)}$. Thus $\{1'\} \cup T \xrightarrow{a^{m-1}} \{1', 1\} \cup (S \setminus \{1\}) = \{1'\} \cup S$ is reachable.

Suppose $q = 1 + 2k$ is the smallest element of S. By Lemma 2, since $S \subseteq \{1 + 2k, \dots, n-1\}$, each element of S has an unique inverse under $(ab)^k$. Thus the size of $S(ab)^{-k}$ has the same size as S. Furthermore, $q - 2k = 1$ is the inverse of q under $(ab)^k$, and thus $1 \in S(ab)^{-k}$. In particular, 1 is the smallest element of $S(ab)^{-k}$, so we can reach $\{1'\} \cup S(ab)^{-k}$. Also, $1'$ is fixed by ab. Thus $\{1'\} \cup S(ab)^{-k} \xrightarrow{(ab)^k} \{1'\} \cup S$ is reachable.

This shows $\{1'\} \cup S$ is reachable whenever the smallest element of S is odd. We deal with the case where the smallest element of S is even in two steps. First, we reduce to the case where S contains at least one odd element. Assume that $\{1'\} \cup T$ is reachable whenever T contains at least one odd element and $|T| \leqslant |S|$, and assume S contains no odd elements. Let $q = 2s$ be the smallest

element of S. Observe that $S \setminus \{q\} \subseteq \{1 + 2s, \ldots, n - 1\}$ and thus by Lemma 2, the set $(S \setminus \{q\})(ab)^{-s}$ has the same size as $S \setminus \{q\}$.

If $n - 1$ is odd, we can reach $\{1', n - 1\} \cup (S \setminus \{q\})(ab)^{-s}$, since $\{n - 1\} \cup (S \setminus \{q\})(ab)^{-s}$ has the same size as S and contains an odd element. Then since $n - 1 \xrightarrow{ab} 2 \xrightarrow{(ab)^{s-1}} 2 + 2(s - 1) = 2s = q$, applying $(ab)^s$ gives $\{1'\} \cup S$.

If $n - 1$ is even, we can reach $\{1', n - 2\} \cup (S \setminus \{q\})(ab)^{-s}$. Since $n - 2 \xrightarrow{a} n - 1 \xrightarrow{b} 2$, we have $n - 2 \xrightarrow{ab} 2 \xrightarrow{(ab)^{s-1}} 2s = q$ and thus applying $(ab)^s$ gives $\{1'\} \cup S$.

Finally, we deal with the case where S contains at least one odd element. Let p be the smallest odd element of S. Let $T = \{q \in S : p \leqslant q\}$, the set of elements above (and including) the smallest odd element. Thus $S \setminus T$ is the set of elements of S that are strictly below the smallest odd element (in particular, each element of $S \setminus T$ is even). We prove that $\{1'\} \cup S$ is reachable by induction on the size of $S \setminus T$. This is the *secondary claim*.

The base case is $|S \setminus T| = 0$. Here every element of S is greater than or equal to p. Thus we are in the case where the smallest element of S is odd, which we have solved.

Suppose that whenever $|S \setminus T| < k$, we can reach $\{1'\} \cup S$. That is, whenever there are fewer than k elements strictly below the smallest odd element of S, we can reach $\{1'\} \cup S$. This is the *secondary induction hypothesis*. We want to prove the secondary claim holds when $|S \setminus T| = k$ for $k \geqslant 1$.

Let r be the smallest element of S. Since $k \geqslant 1$, we have $r \in S \setminus T$ and thus r is even. Suppose $r = 2s$. Then $S \setminus \{r\} \subseteq \{1 + 2s, \ldots, n - 1\}$, so by Lemma 2, every element of $S \setminus \{r\}$ has a unique inverse under $(ab)^s$.

In S, the number of elements strictly below the smallest odd element is $|S \setminus T| = k$. Say a set U is *nice* if the number of elements strictly below the smallest odd element of U is *strictly less than* k. Consider the set $S \setminus \{r\}$. Since r is strictly below the smallest odd element of S, it follows that $S \setminus \{r\}$ is nice. If we apply $(ab)^{-s}$ to $S \setminus \{r\}$, this simply subtracts $2s$ from every element of $S \setminus \{r\}$, and so preserves parity. Thus $(S \setminus \{r\})(ab)^{-s}$ is also nice. Finally, if we take the union $(S \setminus \{r\})(ab)^{-s} \cup \{n - 1\}$, this set is nice, since $n - 1$ is the largest element of this set and so cannot possibly be strictly below anything.

The secondary induction hypothesis says precisely that $\{1'\} \cup U$ is reachable whenever U is nice. Thus $\{1', n - 1\} \cup (S \setminus \{r\})(ab)^{-s}$ is reachable. Notice that $n - 1 \xrightarrow{(ab)^s} 2s = r$. Hence by $(ab)^s$ we reach $\{1'\} \cup S$.

This proves the *secondary claim*. We have thus shown that if each state $\{p'\} \cup T$ with $|T| < |S|$ is reachable, then each state $\{1'\} \cup S$ is reachable. To see that each $\{p'\} \cup S$ is reachable, reach $\{1'\} \cup Sa^{-(p-1)}$ and apply a^{p-1} to reach $\{p'\} \cup S$. This completes the proof of the *primary claim*, and shows that $(m - 1)2^{n-1} + 1$ states are reachable.

Now, we show that all reachable states are distinguishable. Let $\{p'\} \cup S$ and $\{q'\} \cup T$ be two distinct states. Suppose $S \neq T$ and suppose $r \in S \oplus T$, the symmetric difference of S and T. Then a^{n-1-r} distinguishes the states.

Now suppose $S = T$ and $p' \neq q'$. Assume without loss of generality that $p < q$. Define $U = Sa^{m-1-q}$. Then we have

$$\{q'\} \cup S \xrightarrow{a^{m-1-q}} \{(m-1)'\} \cup U \xrightarrow{b} \{(m-1)', 1\} \cup Ub.$$

Observe that $(Q_n \setminus \{0\})b = \{2, \ldots, n-1\}$. Thus $Ub \subseteq \{2, \ldots, n-1\}$ and in particular, $Ub \cap \{1\} = \emptyset$.

On the other hand, let $r' = (p+m-1-q)'$. If we apply $a^{m-1-q}b$ to $\{p'\} \cup S$, we reach

$$\{p'\} \cup S \xrightarrow{a^{m-1-q}} \{r'\} \cup U \xrightarrow{b} \{r'\}b \cup Ub.$$

Note that since $p < q$, we have $r = p + m - 1 - q < m - 1$. Hence b either fixes r', or if $r' \in \{1', 2'\}$, it swaps r' with the other element of $\{1', 2'\}$. In either case $r'b < m - 1$. It follows that $\{r'\}b = \{r'b\}$.

Thus $\{p'\} \cup S$ is sent to $\{r'b\} \cup Ub$ by $a^{m-1-q}b$, and $\{q'\} \cup S$ is sent to $\{(m-1)'\} \cup (\{1\} \cup Ub)$, with $Ub \cap \{1\} = \emptyset$. These two states have different subsets of Q_n, so they are distinguishable. This proves that all states are distinguishable, and hence the bound is met. □

Proposition 10 (Unrestricted Product). *For $m, n \geqslant 4$, let L'_m (respectively, L_n) be a non-returning language of complexity m (respectively, n) over an alphabet Σ', (respectively, Σ). Then the complexity of product is at most $m2^{n+1} + 1$, and this bound is met by $L'_m(a, b)$ and $L_n(a, -, b, d)$.*

Here $L_n(a, -, b, d)$ is the language of DFA $\mathcal{D}_n(a, -, b, d)$ with transformations $a\colon (1, \ldots, n-1)(0 \to 1)$, $b\colon (2, \ldots, n-1)(1{\to}2)(0{\to}1)$ and $d\colon (0{\to}2)$. That is, everything is the same as DFA $\mathcal{D}_n(a, b, c, d)$, except c is removed and b now induces the transformation that c originally induced.

Proof. First we derive an upper bound; the derivation is similar to that for regular languages [3]. Let $\mathcal{D}'_m = (Q'_m, \Sigma', \delta', 0', F')$ and $\mathcal{D}_n = (Q_n, \Sigma, \delta, 0, F)$ be minimal DFAs of non-returning languages $L'_m \subseteq (\Sigma')^*$ and $L_n \subseteq \Sigma^*$, respectively. We use the same construction as in the restricted case. As in that case, we may possibly reach the initial state $\{0'\}$ and $(m-1)2^{n-1}$ states using letters in $\Sigma' \cap \Sigma$. But starting in any set of the form $\{p'\} \cup S$, where $S \subseteq \{1, \ldots, n-1\}$, and applying a letter from $\Sigma \setminus \Sigma'$ we may also reach S. Hence at most $(m-1)2^{n-1} + 2^{n-1} + 1 = m2^{n-1} + 1$ states can be reached.

Languages $L'_m(a, b)$ and $L_n(a, -, b, d)$ meet this bound. We can reach $(m-1)2^{n-1} + 1$ states by words in $\{a, b\}^*$. From $\{p'\} \cup S$, we can use d as described above to reach S and thus reach 2^{n-1} more states. It remains to show that all these states are distinguishable.

We will say a state is a Q'_m-state if it contains an element of Q'_m, and a Q_n-state otherwise. The reachable Q'_m-states have the form $\{p'\} \cup S$ with $p' \in Q'_m \setminus \{0'\}$ and $S \subseteq Q_n$. The reachable Q_n-states have the form $S \subseteq Q_n$.

All Q'_m-states are all pairwise distinguishable using words over $\{a, b\}^*$, using the same arguments as the restricted case. Given two distinct Q_n-states S and T, we can take $r \in S \oplus T$ and distinguish the states using a^{n-1-r}, as in the restricted case.

Now let $\{p'\} \cup S$ be a Q'_m-state and let T be a Q_n-state. If $S \neq T$, we can take $r \in S \oplus T$ and use a^{n-1-r} to distinguish. If $S = T$, let $U = Sa^{m-1-p}$ so that $\{p'\} \cup S \xrightarrow{a^{m-1-p}} \{(m-1)'\} \cup U \xrightarrow{b} \{(m-1)', 1\} \cup Ub$. As in the restricted case, we have $Ub \subseteq \{2, \ldots, n-1\}$. So $a^{m-1-p}b$ sends $\{p'\} \cup S$ to $\{(m-1)'\} \cup (\{1\} \cup Ub)$, with $Ub \cap \{1\} = \emptyset$, but it sends S simply to Ub. These two states have different subsets of Q_n, so they are distinguishable.

So all Q'_m states are pairwise distinguishable, all Q_n-states are pairwise distinguishable, and all Q'_m-states are distinguishable from all Q_n-states; hence all states are distinguishable. □

8   Boolean Operations

Proposition 11 (Restricted Boolean Operations). *Let $\mathcal{D}_n(a,b)$ be the DFA of Definition 1 and let $L_n(a,b)$ be its language. Then for $m, n \geqslant 4$ and for any proper binary boolean operation $\circ$ the complexity of $L'_m(a,b) \circ L_n(b,a)$ is $mn - (m + n - 2)$. If $m \neq n$ then $\kappa(L'_m(a,b) \circ L_n(a,b)) = mn - (m + n - 2)$.*

Proof. The upper bound was established in [9]. For the lower bound, Figure 5 shows the two argument DFAs. As usual we construct their direct product. State $(0', 0)$ is initial and can never be reached again. If we apply a, we reach state $(1', 2)$, and the states reachable from this state form the direct product of DFA $\mathcal{E}'_{m-1}(a,b) = (\{1, \ldots, (m-1)'\}, \{a, b\}, \delta', 1', \{(m-1)'\})$ and DFA $\mathcal{E}_{n-1}(b,a) = (\{1, \ldots, n-1\}, \{a, b\}, \delta, 2, \{n-1\})$, where δ' and δ are $\delta_{m'}$ and δ_n restricted to $Q'_m \setminus \{0'\}$ and $Q_n \setminus \{0\}$. Since the transition semigroups of $\mathcal{E}'_m$ and $\mathcal{E}_n$ are the symmetric groups S_m and S_n, respectively, the result from [1, Theorem 1] applies, except in the cases where (m, n) is in $\{(4,5), (5,4), (5,5)\}$, which have been verified by computation. Our first claim follows for the remaining cases by [1, Theorem 1]. If $m \neq n$, [1, Theorem 1] applies to $\mathcal{D}'_m(a,b)$ and $\mathcal{D}_n(a,b)$, and the second claim follows. In both cases the direct product of $\mathcal{E}'_m$ and $\mathcal{E}_n$ has $(m-1)(n-1)$ states; hence in the direct product of $\mathcal{D}'_m$ and $\mathcal{D}_n$ there are $(m-1)(n-1) + 1 = mn - (m+n-2)$ states. By [1, Theorem 1] all these states are reachable and pairwise distinguishable for every proper operation $\circ$.

Finally, note also that $(0', 0)$ is distinguishable from all other states. Since $(0', 0)a = (1', 2)$ and $(1', 2)a^{-1} = \{(m-1)', 1)\}$, we see that $(0', 0)$ is distinguishable from $(p', q) \neq ((m-1)', 1)$ by first applying a, then applying a word that distinguishes $(0', 0)a = (1', 2)$ from $(p', q)a$. It is distinguishable from $((m-1)', 1)$ by first applying b, then applying a word that distinguishes $(0', 0)b = (2', 1)$ from $((m-1)', 1)b = ((m-1)', 2)$. □

Proposition 12 ((Unrestricted Boolean Operations)). *For $m, n \geqslant 4$, let $L'_m(\Sigma')$ (respectively, $L_n(\Sigma)$) be a non-returning language of complexity m (respectively, n) over an alphabet Σ', (respectively, Σ). Then the complexity of union and symmetric difference is $mn + 1$ and this bound is met by $L'_m(a, b, c)$ and $L_n(b, a, d)$; the complexity of difference is $mn - n + 1$, and this bound is met*
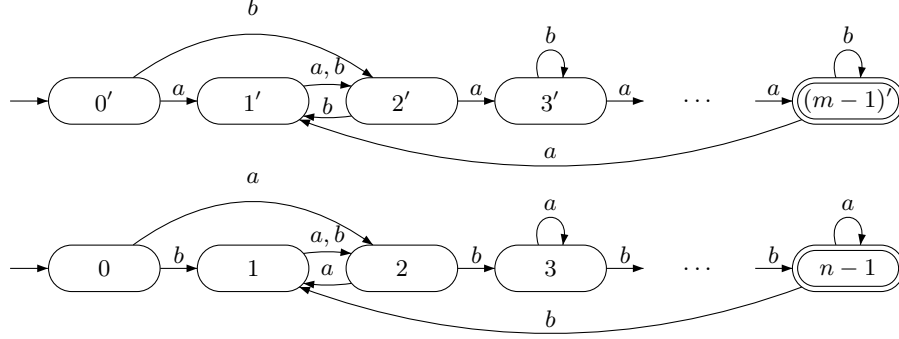
Fig. 5. DFAs $\mathcal{D}'_m(a, b)$ and $\mathcal{D}_n(b, a)$ for boolean operations.

by $L'_m(a, b, c)$ and $L_n(b, a)$; the complexity of intersection is $mn - (m + n - 2)$ and this bound is met by $L'_m(a, b)$ and $L_n(b, a)$.

Proof. First we derive upper bounds for four boolean operations. Let $\mathcal{D}'_m(\Sigma') = (Q'_m, \Sigma', \delta'_m, 0', F')$ and $\mathcal{D}_n(\Sigma) = (Q_n, \Sigma, \delta_n, 0, F)$ be minimal DFAs for non-returning languages $L'_m(\Sigma')$ and $L_n(\Sigma)$ of complexity m and n, respectively. Assume that $\Sigma' \setminus \Sigma$ and $\Sigma \setminus \Sigma'$ are non-empty; this assumption leads to the largest upper bound. We add an empty state $\emptyset'$ to $\mathcal{D}'_m$ to send all transitions under the letters from $\Sigma \setminus \Sigma'$ to that state; thus we get an $(m + 1)$-state DFA $\mathcal{D}'_{m,\emptyset'}$. Similarly, we add an empty state $\emptyset$ to $\mathcal{D}_n$ to get $\mathcal{D}_{n,\emptyset}$. Now we construct the direct product of $\mathcal{D}'_{m,\emptyset'}$ and $\mathcal{D}_{n,\emptyset}$. There are at most $(m + 1)(n + 1)$ states in this direct product. Note, however, that the m states of the form $(p', 0)$, $p \neq 0$, and the n states of the form $(0', q)$, $q \neq 0$ are not reachable. Thus we have an upper bound of $(m + 1)(n + 1) - (m + n) = mn + 1$. We will show that this bound can be reached for union and symmetric difference. However, the alphabet of the difference $L'_{m,\emptyset'} \setminus L_{n,\emptyset}$ is Σ' and states of the form $(\emptyset', q)$, $q \in \{1, \ldots, n - 1\} \cup \{\emptyset\}$, which are reachable only by letters in $\Sigma \setminus \Sigma'$, are not reachable in the difference DFA. Hence the bound reduces to $mn - n + 1$. Similarly, the alphabet of the intersection $L'_{m,\emptyset'} \cap L_{n,\emptyset}$ is $\Sigma' \cap \Sigma$ and states of the form $(p', \emptyset)$, $p \in \{1', \ldots, (m - 1)'\} \cup \{\emptyset'\}$, $(\emptyset', q)$, $q \in \{1, \ldots, n - 1\} \cup \{\emptyset\}$ are not reachable. Hence the bound reduces to $mn - (m + n - 2)$, which is the same as the bound in the restricted case.

We now claim that our witnesses of Figure 6 meet this bound.
Union Consider the direct product of $\mathcal{D}'_m$ and $\mathcal{D}_n$. State $(0', 0)$ is initial and reaches $(1', 2)$ by a. By [1, Theorem 1] and computation in the cases where (m, n) is in $\{(4, 5), (5, 4), (5, 5)\}$ all $(m - 1)(n - 1)$ states of the form (p', q), $p' \notin \{0', \emptyset'\}$, and $q \notin \{\emptyset, 0\}$ are reachable and pairwise distinguishable. The states that have an empty component are reachable as follows: $(\emptyset', 1)$ can reached from any reachable state $(q', 1)$ by d, and $(\emptyset', 1) \xrightarrow{b^{q-1}} (\emptyset', q)$, for $q = 1, \ldots, n - 1$. The state $(1', \emptyset)$ is reached from $(0', 0)$ by c, and $(1', \emptyset) \xrightarrow{a^{p-1}} (p', \emptyset)$, for $p =$
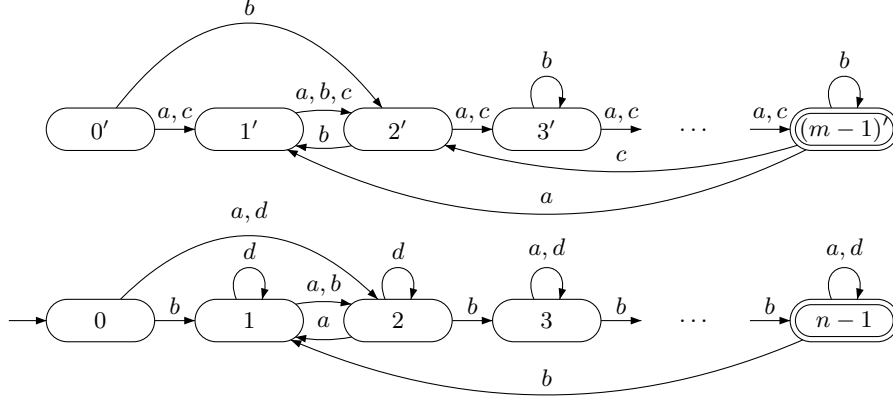
Fig. 6. DFAs $\mathcal{D}'_m(a, b, c)$ and $\mathcal{D}_n(b, a, d)$ for unrestricted boolean operations.

$1, \ldots, m-1$. Finally, $(\emptyset', \emptyset)$ is reached from any state $(\emptyset', q)$ by c. Thus we have a total of $1 + (m-1)(n-1) + (m-1) + (n-1) + 1 = mn + 1$ states.

Let $S = \{(p', q) \mid 1 \leqslant p \leqslant m-1, 1 \leqslant q \leqslant n-1\}$, $V = \{(p', \emptyset) \mid 1 \leqslant p \leqslant m-1\}$, and $H = \{(\emptyset', q) \mid 1 \leqslant q \leqslant n-1\}$. States in H are distinguishable by words in b^*, and those in V, by words in a^*. State $(\emptyset', \emptyset)$ is the only empty state; hence it is distinguishable from every other reachable state. State $(0', 0)$ is distinguished from every other reachable non-final state because only it accepts a^{m-1}. Non-final states in H are distinguished from those in S and V by words in a^*. Non-final states in V are distinguished from those in S by words in b^*. State $(\emptyset', n-1)$ is distinguished from $((m-1)', \emptyset)$ by d, and from final states in S by words in ca^*. State $((m-1)', \emptyset)$ is distinguished from final states in S by words in db^*.

Symmetric Difference Now state $((m-1)', n-1)$ is no longer final, but the rest of the argument is the same as for union.

Difference Here states $(p', n-1)$, $p' \in \{1', \ldots, (m-1)'\} \cup \{\emptyset'\}$ are no longer final. Since the alphabet of $L'_m \setminus L_n$ is Σ', states in H are not reachable, and there are only $mn - n + 1$ reachable states. The states in S are pairwise distinguishable by [1, Theorem 1] and those in V by the argument used for union. The arguments used for union also apply here to distinguish states in S from those in V.

Intersection Now the alphabet of $L'_m \cap L_n$ is $\Sigma' \cap \Sigma$, and neither the states in H nor those in V are reachable. The claim now follows by [1, Theorem 1].   □

The complexity of any other binary boolean operation can be determined from the complexities of union, intersection, difference and symmetric difference; however, the complexity of $L_m \circ L'_n$ may differ by 1 from the complexity of $\overline{L_m \circ L'_n}$. For more details see [5].

9   Conclusions

We have shown that there exists a most complex non-returning language stream $(L_4(\Sigma), \dots, L_n(\Sigma), \dots)$. The cardinality of the syntactic semigroup of $L_n(\Sigma)$ is $(n-1)^n$ and its atoms are have the highest state complexity possible for non-returning languages; both of these bounds can be reached only if Σ has at least $\binom{n}{2}$ letters. The bounds for the common restricted operations, however, can be met by streams over $\{a, b, c\}$ or $\{a, b\}$: $\kappa(L_m(a, b) \circ L_n(b, a)) = mn - (m + n - 2)$ for all proper boolean operations $\circ$; $\kappa(L_n(a, b))^* = 2^{n-1}$; $\kappa(L_n(a, b, c)^R) = 2^n$; and $\kappa(L'_m(a, b)L_n(a, -, b)) = (m - 1)2^{n-1} + 1$. The bounds for unrestricted boolean operations can be met by $L'_m(a, b, c)$ and $L_n(b, a, d)$, whereas those for the unrestricted product, by $L'_m(a, b)$ and $L_n(a, -, b, d)$.

References

1. Bell, J., Brzozowski, J.A., Moreira, N., Reis, R.: Symmetric groups and quotient complexity of boolean operations. In: Esparza, J., et al. (eds.) ICALP 2014. LNCS, vol. 8573, pp. 1–12. Springer (2014)
2. Brzozowski, J.A.: In search of the most complex regular languages. Int. J. Found. Comput. Sc. 24(6), 691–708 (2013)
3. Brzozowski, J.A.: Unrestricted state complexity of binary operations on regular languages. In: C. Câmpeanu et al. (ed.) DCFS. LNCS, vol. 9777, pp. 60–72. Springer (2016)
4. Brzozowski, J.A., Davies, S.: Quotient complexities of atoms of regular ideal languages. Acta Cybernet. 22, 293–311 (2015)
5. Brzozowski, J.A., Sinnamon, C.: Unrestricted state complexity of binary operations on regular and ideal languages (2016), http://arxiv.org/abs/1609.04439
6. Brzozowski, J.A., Tamm, H.: Complexity of atoms of regular languages. Int. J. Found. Comput. Sc. 24(7), 1009–1027 (2013)
7. Brzozowski, J.A., Tamm, H.: Theory of átomata. Theoret. Comput. Sci. 539, 13–27 (2014)
8. Brzozowski, J.A., Ye, Y.: Syntactic complexity of ideal and closed languages. In: Mauri, G., Leporati, A. (eds.) DLT. LNCS, vol. 6795, pp. 117–128. Springer (2011)
9. Eom, H.S., Han, Y.S., Jirásková, G.: State complexity of basic operations on non-returning regular languages. Fund. Inform. 144, 161–182 (2016)
10. Holzer, M., König, B.: On deterministic finite automata and syntactic monoid size. Theoret. Comput. Sci. 327, 319–347 (2004)
11. Iván, S.: Complexity of atoms, combinatorially. Inform. Process. Lett. 116(5), 356–360 (2016)
12. Krawetz, B., Lawrence, J., Shallit, J.: State complexity and the monoid of transformations of a finite set. In: Domaratzki, M., Okhotin, A., Salomaa, K., Yu, S. (eds.) Proceedings of the Implementation and Application of Automata, (CIAA). LNCS, vol. 3317, pp. 213–224. Springer (2005)
13. Perrin, D.: Finite automata. In: van Leewen, J. (ed.) Handbook of Theoretical Computer Science, vol. B, pp. 1–57. Elsevier (1990)

14. Pin, J.E.: Syntactic semigroups. In: Handbook of Formal Languages, vol. 1: Word, Language, Grammar, pp. 679–746. Springer, New York, NY, USA (1997)
15. Yu, S.: Regular languages. In: Rozenberg, G., Salomaa, A. (eds.) Handbook of Formal Languages, pp. 41–110. Springer (1997)