Complexity of Proper Prefix-Convex Regular Languages

Janusz A. Brzozowski and Corwin Sinnamon(✉)

David R. Cheriton School of Computer Science, University of Waterloo,
Waterloo, ON N2L 3G1, Canada
brzozo@uwaterloo.ca, sinncore@gmail.com

Abstract. A language L over an alphabet Σ is prefix-convex if, for any words $x, y, z \in \Sigma^*$, whenever x and xyz are in L, then so is xy. Prefix-convex languages include right-ideal, prefix-closed, and prefix-free languages, which were studied elsewhere. Here we concentrate on prefix-convex languages that do not belong to any one of these classes; we call such languages *proper*. We exhibit most complex proper prefix-convex languages, which meet the bounds for the size of the syntactic semigroup, reversal, complexity of atoms, star, product, and Boolean operations.

Keywords: Atom · Most complex · Prefix-convex · Proper · Quotient complexity · Regular language · State complexity · Syntactic semigroup

1 Introduction

Prefix-Convex Languages. We examine the complexity properties of a class of regular languages that has never been studied before: the class of proper prefix-convex languages [7]. Let Σ be a finite alphabet; if $w = xy$, for $x, y \in \Sigma^*$, then x is a prefix of w. A language $L \subseteq \Sigma^*$ is *prefix-convex* [1,16] if whenever x and xyz are in L, then so is xy. Prefix-convex languages include three special cases:

1. A language $L \subseteq \Sigma$ is a *right ideal* if it is non-empty and satisfies $L = L\Sigma^*$. Right ideals appear in pattern matching [11]: $L\Sigma^*$ is the set of all words in some text (word in Σ^*) beginning with words in L.
2. A language is *prefix-closed* [6] if whenever w is in L, then so is every prefix of w. The set of allowed sequences to any system is prefix-closed. Every prefix-closed language other than Σ^* is the complement of a right ideal [1].
3. A language is *prefix-free* if $w \in L$ implies that no prefix of w other than w is in L. Prefix-free languages other than $\{\varepsilon\}$, where ε is the empty word, are prefix codes and are of considerable importance in coding theory [2].

This work was supported by the Natural Sciences and Engineering Research Council of Canada grant no. OGP0000871.

© Springer International Publishing AG 2017
A. Carayol and C. Nicaud (Eds.): CIAA 2017, LNCS 10329, pp. 52–63, 2017.
DOI: 10.1007/978-3-319-60134-2_5

The complexities of these three special prefix-convex languages were studied in [8]. We now turn to the "real" prefix-convex languages that do not belong to any of the three special classes.

Omitted proofs can be found in [7].

Complexities of Operations. If $L \subseteq \Sigma^*$ is a language, the *(left) quotient* of L by a word $w \in \Sigma^*$ is $w^{-1}L = \{x \mid wx \in L\}$. A language is regular if and only if it has a finite number of distinct quotients. So the number of quotients of L, the *quotient complexity* [3] $\kappa(L)$ of L, is a natural measure of complexity for L. An equivalent concept is the *state complexity* [15,17,18] of L, which is the number of states in a complete minimal deterministic finite automaton (DFA) over Σ recognizing L. We refer to quotient/state complexity simply as *complexity*.

If L_n is a regular language of complexity n, and $\circ$ is a unary operation, the *complexity of* $\circ$ is the maximal value of $\kappa(L_n^\circ)$, expressed as a function of n, as L_n ranges over all languages of complexity n. If L'_m and L_n are regular languages of complexities m and n respectively, and $\circ$ is a binary operation, the *complexity of* $\circ$ is the maximal value of $\kappa(L'_m \circ L_n)$, expressed as a function of m and n, as L'_m and L_n range over all languages of complexities m and n. The complexity of an operation is a lower bound on its time and space complexities. The operations reversal, (Kleene) star, product (concatenation), and binary boolean operations are considered "common", and their complexities are known; see [4,17,18].

Witnesses. To find the complexity of a unary operation we find an upper bound on this complexity, and languages that meet this bound. We require a language L_n for each n, that is, a sequence, $(L_k, L_{k+1}, \dots)$, called a *stream* of languages, where k is a small integer, because the bound may not hold for small values of n. For a binary operation we need two streams. The same stream cannot always be used for both operands, but for all common binary operations the second stream can be a "dialect" of the first, that is it can "differ only slightly" from the first [4]. Let $\Sigma = \{a_1, \dots, a_k\}$ be an alphabet ordered as shown; if $L \subseteq \Sigma^*$, we denote it by $L(a_1, \dots, a_k)$. A *dialect* of L is obtained by deleting letters of Σ in the words of L, or replacing them by letters of another alphabet Σ'. More precisely, for an injective partial map $\pi\colon \Sigma \mapsto \Sigma'$, we get a dialect of L by replacing each letter $a \in \Sigma$ by $\pi(a)$ in every word of L, or deleting the word if $\pi(a)$ is undefined. We write $L(\pi(a_1), \dots, \pi(a_k))$ to denote the dialect of $L(a_1, \dots, a_k)$ given by π, and we denote undefined values of π by "$-$". Undefined values for letters at the end of the alphabet are omitted; for example, $L(a, c, -, -)$ is written as $L(a, c)$. Our definition of dialect is more general than that of [5], where only the case $\Sigma' = \Sigma$ was allowed.

Finite Automata. 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 set of all words accepted by $\mathcal{D}$ is the *language of* $\mathcal{D}$. If $q \in Q$, then the *language* L_q *of* q is the language accepted by the DFA $(Q, \Sigma, \delta, q, F)$. A state is *empty or dead or a sink* 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*. An ε-*NFA* is an NFA in which transitions under the empty word ε are also permitted.

Transformations. We use $Q_n = \{0, \dots, n-1\}$ as the set of states of every DFA with n states. 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 qt. In any DFA, each letter $a \in \Sigma$ induces a transformation δ_a of the set Q_n defined by $q\delta_a = \delta(q, a)$; we denote this by $a\colon \delta_a$. Often we use the letter a to denote the transformation it induces; thus we write qa instead of $q\delta_a$. We extend the notation to sets: if $P \subseteq Q_n$, then $Pa = \{pa \mid p \in P\}$. We also write $P \xrightarrow{a} Pa$ to indicate that the image of P under a is Pa. If s, t are transformations of Q_n, their composition is $(qs)t$.

For $k \geqslant 2$, a transformation (permutation) t of a set $P = \{q_0, q_1, \dots, q_{k-1}\} \subseteq Q_n$ is a k-*cycle* if $q_0 t = q_1, q_1 t = q_2, \dots, q_{k-2} t = q_{k-1}, q_{k-1} t = q_0$. This k-cycle is denoted by $(q_0, q_1, \dots, q_{k-1})$. A 2-cycle (q_0, q_1) is called a *transposition*. A transformation that sends all the states of P to q and acts as the identity on the other states is denoted by $(P \to q)$, and $(Q_n \to p)$ is called a *constant* transformation. If $P = \{p\}$ we write $(p \to q)$ for $(\{p\} \to q)$. The identity transformation is denoted by $\mathbb{1}$. Also, $(_i^j\ q \to q+1)$ is a transformation that sends q to $q+1$ for $i \leqslant q \leqslant j$ and is the identity for the remaining states; $(_i^j\ q \to q-1)$ is defined similarly.

Semigroups. The *syntactic congruence* of $L \subseteq \Sigma^*$ is defined on Σ^+: 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 the *syntactic semigroup* of L. Let $\mathcal{D}_n = (Q_n, \Sigma, \delta, q_0, F)$ be a DFA, and let $L_n = L(\mathcal{D}_n)$. 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 $T_{\mathcal{D}_n}$ of all such transformations by non-empty words is a semigroup under composition called the *transition semigroup* of $\mathcal{D}_n$. If $\mathcal{D}_n$ is a minimal DFA of L_n, then $T_{\mathcal{D}_n}$ is isomorphic to the syntactic semigroup T_{L_n} of L_n, and we represent elements of T_{L_n} by transformations in $T_{\mathcal{D}_n}$. The size of the syntactic semigroup has been used as a measure of complexity for regular languages [4,10,12,14].

Atoms. are defined by a left congruence, where two words x and y are equivalent if $ux \in L$ if and only if $uy \in L$ for all $u \in \Sigma^*$. Thus x and y are equivalent if $x \in u^{-1}L$ if and only if $y \in u^{-1}L$. An equivalence class of this relation is an *atom* of L [9,13].

One can conclude that an atom is a non-empty intersection of complemented and uncomplemented quotients of L. That is, every atom of a language with quotients $K_0, K_1, \dots, K_{n-1}$ can be written as $A_S = \bigcap_{i \in S} K_i \cap \bigcap_{i \in \overline{S}} \overline{K_i}$ for some set $S \subseteq Q_n$. The number of atoms and their complexities were suggested as possible measures of complexity [4], because all the quotients of a language and the quotients of its atoms are unions of atoms [9].

Most Complex Regular Stream. The stream $(\mathcal{D}_n(a,b,c) \mid n \geqslant 3)$ of Definition 1 and Fig. 1 will be used as a component in the class of proper prefix-convex languages. This stream together with some dialects meets the complexity bounds for reversal, star, product, and all binary boolean operations [7,8]. Moreover, it has the maximal syntactic semigroup and most complex atoms, making it a most complex regular stream.

Definition 1. *For $n \geqslant 3$, let $\mathcal{D}_n = \mathcal{D}_n(a,b,c) = (Q_n, \Sigma, \delta_n, 0, \{n-1\})$, where $\Sigma = \{a,b,c\}$, and δ_n is defined by a: $(0,\dots,n-1)$, b: $(0,1)$, c: $(1 \to 0)$.*

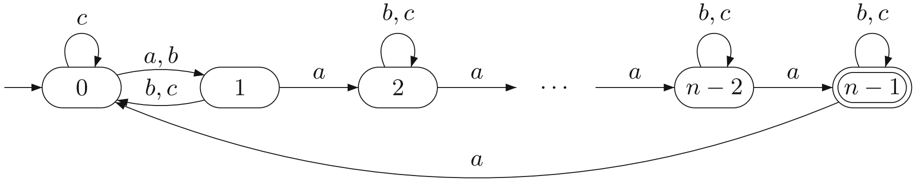

Fig. 1. Minimal DFA of a most complex regular language.

Most complex streams are useful in systems dealing with regular languages and finite automata. To know the maximal sizes of automata that can be handled by a system it suffices to use the most complex stream to test all the operations.

2 Proper Prefix-Convex Languages

We begin with some properties of prefix-convex languages that will be used frequently in this section. The following lemma and propositions characterize the classes of prefix-convex languages in terms of their minimal DFAs.

Lemma 1. *Let L be a prefix-convex language over Σ. Either L is a right ideal or L has an empty quotient.*

Proposition 1. *Let L_n be a regular language of complexity n, and let $\mathcal{D}_n = (Q_n, \Sigma, \delta, 0, F)$ be a minimal DFA recognizing L_n. The following are equivalent:*

1. *L_n is prefix-convex.*
2. *For all $p,q,r \in Q_n$, if p and r are final, q is reachable from p, and r is reachable from q, then q is final.*
3. *Every state reachable in $\mathcal{D}_n$ from any final state is either final or empty.*

Proposition 2. *Let L_n be a non-empty prefix-convex language of complexity n, and let $\mathcal{D}_n = (Q_n, \Sigma, \delta, 0, F)$ be a minimal DFA recognizing L_n.*

1. *L_n is prefix-closed if and only if $0 \in F$.*
2. *L_n is prefix-free if and only if $\mathcal{D}_n$ has a unique final state p and an empty state p' such that $\delta(p,a) = p'$ for all $a \in \Sigma$.*
3. *L_n is a right ideal if and only if $\mathcal{D}_n$ has a unique final state p and $\delta(p,a) = p$ for all $a \in \Sigma$.*

A prefix-convex language L is *proper* if it is not a right ideal and it is neither prefix-closed nor prefix-free. We say it is *k-proper* if it has k final states, $1 \leqslant k \leqslant n-2$. Every minimal DFA for a k-proper language with complexity n has the same general structure: there are $n-1-k$ non-final, non-empty states, k final states, and one empty state. Every letter fixes the empty state and, by Proposition 1, no letter sends a final state to a non-final, non-empty state.

Next we define a stream of k-proper DFAs and languages, which we will show to be most complex.

Definition 2. *For $n \geqslant 3$, $1 \leqslant k \leqslant n-2$, let $\mathcal{D}_{n,k}(\Sigma) = (Q_n, \Sigma, \delta_{n,k}, 0, F_{n,k})$ where $\Sigma = \{a, b, c_1, c_2, d_1, d_2, e\}$, $F_{n,k} = \{n-1-k, \dots, n-2\}$, and $\delta_{n,k}$ is given by the transformations*

$$a\colon \begin{cases} (1,\dots,n-2-k)(n-1-k,n-k), & \text{if } n-1-k \text{ is even and } k \geqslant 2;\\ (0,\dots,n-2-k)(n-1-k,n-k), & \text{if } n-1-k \text{ is odd and } k \geqslant 2;\\ (1,\dots,n-2-k), & \text{if } n-1-k \text{ is even and } k = 1;\\ (0,\dots,n-2-k), & \text{if } n-1-k \text{ is odd and } k = 1. \end{cases}$$

$$b\colon \begin{cases} (n-k,\dots,n-2)(0,1), & \text{if } k \text{ is even and } n-1-k \geqslant 2;\\ (n-1-k,\dots,n-2)(0,1), & \text{if } k \text{ is odd and } n-1-k \geqslant 2;\\ (n-k,\dots,n-2), & \text{if } k \text{ is even and } n-1-k = 1;\\ (n-1-k,\dots,n-2), & \text{if } k \text{ is odd and } n-1-k = 1. \end{cases}$$

$$c_1\colon \begin{cases} (1 \to 0), & \text{if } n-1-k \geqslant 2;\\ \mathbb{1}, & \text{if } n-1-k = 1. \end{cases}$$

$$c_2\colon \begin{cases} (n-k \to n-1-k), & \text{if } k \geqslant 2;\\ \mathbb{1}, & \text{if } k = 1. \end{cases}$$

$$d_1\colon (n-2-k \to n-1)\binom{n-3-k}{0}\ q \to q+1).$$

$$d_2\colon \binom{n-2}{n-1-k}\ q \to q+1).$$

$$e\colon (0 \to n-1-k).$$

Also, let $E_{n,k} = \{0, \dots, n-2-k\}$; it is useful to partition Q_n into $E_{n,k}$, $F_{n,k}$, and $\{n-1\}$. Letters a and b have complementary behaviours on $E_{n,k}$ and $F_{n,k}$, depending on the parities of n and k. Letters c_1 and d_1 act on $E_{n,k}$ in exactly the same way as c_2 and d_2 act on $F_{n,k}$. In addition, d_1 and d_2 send states $n-2-k$ and $n-2$, respectively, to state $n-1$, and letter e connects the two parts of the DFA. The structure of $\mathcal{D}_n(\Sigma)$ is shown in Figs. 2 and 3 for certain parities of $n-1-k$ and k. Let $L_{n,k}(\Sigma)$ be the language recognized by $\mathcal{D}_{n,k}(\Sigma)$.

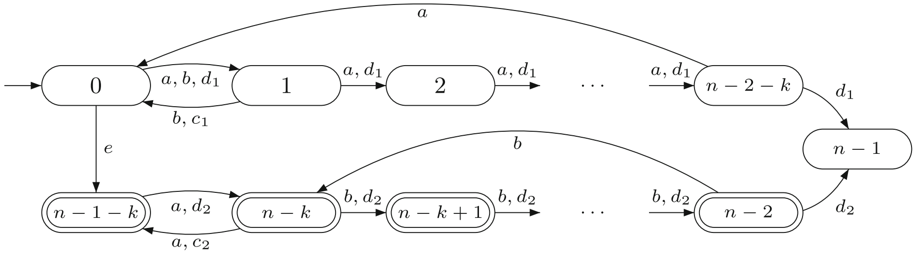

Fig. 2. DFA $\mathcal{D}_{n,k}(a, b, c_1, c_2, d_1, d_2, e)$ of Definition 2 when $n-1-k$ is odd, k is even, and both are at least 2; missing transitions are self-loops.

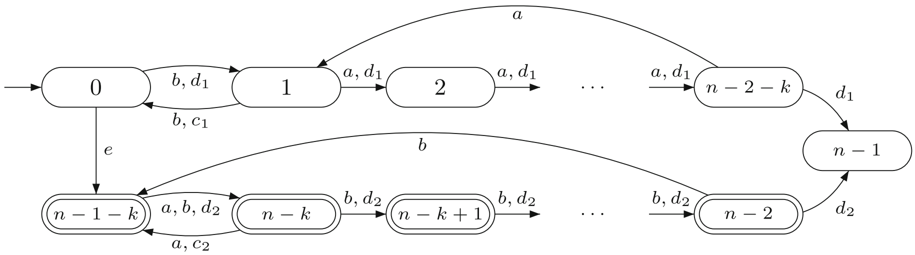

Fig. 3. DFA $\mathcal{D}_{n,k}(a, b, c_1, c_2, d_1, d_2, e)$ of Definition 2 when $n-1-k$ is even, k is odd, and both are at least 2; missing transitions are self-loops.

Theorem 1 (Proper Prefix-Convex Languages). *For $n \geqslant 3$ and $1 \leqslant k \leqslant n-2$, the DFA $\mathcal{D}_{n,k}(\Sigma)$ of Definition 2 is minimal and $L_{n,k}(\Sigma)$ is a k-proper language of complexity n. The bounds below are maximal for k-proper prefix-convex languages. At least seven letters are required to meet these bounds.*

1. *The syntactic semigroup of $L_{n,k}(\Sigma)$ has cardinality $n^{n-1-k}(k+1)^k$; the maximal value $n(n-1)^{n-2}$ is reached only when $k = n-2$.*
2. *The non-empty, non-final quotients of $L_{n,k}(a, b, -, -, -, d_2, e)$ have complexity n, the final quotients have complexity $k+1$, and $\emptyset$ has complexity 1.*
3. *The reverse of $L_{n,k}(a, b, -, -, -, d_2, e)$ has complexity 2^{n-1}; moreover, the language $L_{n,k}(a, b, -, -, -, d_2, e)$ has 2^{n-1} atoms for all k.*
4. *For each atom A_S of $L_{n,k}(\Sigma)$, write $S = X_1 \cup X_2$, where $X_1 \subseteq E_{n,k}$ and $X_2 \subseteq F_{n,k}$. Let $\overline{X_1} = E_{n,k} \setminus X_1$ and $\overline{X_2} = F_{n,k} \setminus X_2$. If $X_2 \neq \emptyset$, then $\kappa(A_S) =$*

$$1 + \sum_{x_1=0}^{|X_1|} \sum_{x_2=1}^{|X_1|+|X_2|-x_1} \sum_{y_1=0}^{|\overline{X_1}|} \sum_{y_2=0}^{|\overline{X_1}|+|\overline{X_2}|-y_1} \binom{n-1-k}{x_1}\binom{k}{x_2}\binom{n-1-k-x_1}{y_1}\binom{k-x_2}{y_2}.$$

If $X_1 \neq \emptyset$ and $X_2 = \emptyset$, then $\kappa(A_S) =$

$$1 + \sum_{x_1=0}^{|X_1|} \sum_{x_2=0}^{|X_1|-x_1} \sum_{y_1=0}^{|\overline{X_1}|} \sum_{y_2=0}^{k} \binom{n-1-k}{x_1}\binom{k}{x_2}\binom{n-1-k-x_1}{y_1}\binom{k-x_2}{y_2} - 2^k \sum_{y=0}^{|\overline{X_1}|} \binom{n-1-k}{y}.$$

Otherwise, $S = \emptyset$ and $\kappa(A_S) = 2^{n-1}$.

5. *The star of* $L_{n,k}(a,b,-,-,d_1,d_2,e)$ *has complexity* $2^{n-2}+2^{n-2-k}+1$*. The maximal value* $2^{n-2}+2^{n-3}+1$ *is reached only when* $k=1$*.*
6. $L'_{m,j}(a,b,c_1,-,d_1,d_2,e)L_{n,k}(a,d_2,c_1,-,d_1,b,e)$ *has complexity* $m-1-j+j2^{n-2}+2^{n-1}$*. The maximal value* $m2^{n-2}+1$ *is reached only when* $j=m-2$*.*
7. *For* $m,n \geqslant 3$*,* $1 \leqslant j \leqslant m-2$*, and* $1 \leqslant k \leqslant n-2$*, define the languages* $L'_{m,j} = L'_{m,j}(a,b,c_1,-,d_1,d_2,e)$ *and* $L_{n,k} = L_{n,k}(a,b,e,-,d_2,d_1,c_1)$*. For any proper binary boolean function* $\circ$*, the complexity of* $L'_{m,j} \circ L_{n,k}$ *is maximal. In particular,*
   (a) $L'_{m,j} \cup L_{n,k}$ *and* $L'_{m,j} \oplus L_{n,k}$ *have complexity* mn*.*
   (b) $L'_{m,j} \setminus L_{n,k}$ *has complexity* $mn-(n-1)$*.*
   (c) $L'_{m,j} \cap L_{n,k}$ *has complexity* $mn-(m+n-2)$*.*

Proof. The remainder of this paper is an outline of the proof of this theorem. The longer parts of the proof are separated into individual propositions and lemmas.

DFA $\mathcal{D}_{n,k}(a,b,-,-,-,d_2,e)$ is easily seen to be minimal. Language $L_{n,k}(\Sigma)$ is k-proper by Propositions 1 and 2.

1. See Lemma 2 and Proposition 3.
2. If the initial state of $\mathcal{D}_{n,k}(a,b,-,-,-,d_2,e)$ is changed to $q \in E_{n,k}$, the new DFA accepts a quotient of $L_{n,k}$ and is still minimal; hence the complexity of that quotient is n. If the initial state is changed to $q \in F_{n,k}$ then states in $E_{n,k}$ are unreachable, but the DFA on $\{n-1-k,\dots,n-1\}$ is minimal; hence the complexity of that quotient is $k+1$. The remaining quotient is empty, and hence has complexity 1. By Proposition 1, these are maximal.
3. See Proposition 4 for the reverse. It was shown in [9] that the number of atoms is equal to the complexity of the reverse.
4. See [7].
5. See Proposition 5.
6. See [7].
7. By [3, Theorem 2], all boolean operations on regular languages have the upper bound mn, which gives the bound for (a). The bounds for (b) and (c) follow from [3, Theorem 5]. The proof that all these bounds are tight for $L'_{m,j} \circ L_{n,k}$ can be found in [7]. $\square$

Lemma 2. *Let* $n \geqslant 1$ *and* $1 \leqslant k \leqslant n-2$*. For any permutation* t *of* Q_n *such that* $E_{n,k}t = E_{n,k}$*,* $F_{n,k}t = F_{n,k}$*, and* $(n-1)t = n-1$*, there is a word* $w \in \{a,b\}^*$ *that induces* t *on* $\mathcal{D}_{n,k}$*.*

Proof. Only a and b induce permutations of Q_n; every other letter induces a properly injective map. Furthermore, a and b permute $E_{n,k}$ and $F_{n,k}$ separately, and both fix $n-1$. Hence every $w \in \{a,b\}^*$ induces a permutation on Q_n such that $E_{n,k}w = E_{n,k}$, $F_{n,k}w = F_{n,k}$, and $(n-1)w = n-1$. Each such permutation naturally corresponds to an element of $S_{n-1-k} \times S_k$, where S_m denotes the symmetric group on m elements. To be consistent with the DFA, assume S_{n-1-k} contains permutations of $\{0,\dots,n-2-k\}$ and S_k contains permutations of $\{n-1-k,\dots,n-2\}$. Let s_a and s_b denote the group elements corresponding to the transformations induced by a and b respectively. We show that s_a and s_b generate $S_{n-1-k} \times S_k$.

It is well known that $(0, \dots, m-1)$, and $(0, 1)$ generate the symmetric group on $\{0, \dots, m-1\}$ for any $m \geq 2$. Note that $(1, \dots, m-1)$ and $(0, 1)$ are also generators, since $(0, 1)(1, \dots, m-1) = (0, \dots, m-1)$.

If $n-1-k = 1$ and $k = 1$, then $S_{n-1-k} \times S_k$ is the trivial group. If $n-1-k = 1$ and $k \geqslant 2$, then $s_a = (\mathbb{1}, (n-1-k, n-k))$ and s_b is either $(\mathbb{1}, (n-1-k, \dots, n-2))$ or $(\mathbb{1}, (n-k, \dots, n-2))$, and either pair generates the group. There is a similar argument when $k = 1$.

Assume now $n-1-k \geqslant 2$ and $k \geqslant 2$. If $n-1-k$ is odd then $s_a = ((0, \dots, n-2-k), (n-1-k, n-k))$, and hence $s_a^{n-1-k} = ((0, \dots, n-2-k)^{n-1-k}, (n-1-k, n-k)^{n-1-k}) = (\mathbb{1}, (n-1-k, n-k))$. Similarly if $n-1-k$ is even then $s_a = ((1, \dots, n-2-k), (n-1-k, n-k))$, and hence $s_a^{n-2-k} = (\mathbb{1}, (n-1-k, n-k))$. Therefore $(\mathbb{1}, (n-1-k, n-k))$ is always generated by s_a. By symmetry, $((0, 1), \mathbb{1})$ is always generated by s_b regardless of the parity of k.

Since we can isolate the transposition component of s_a, we can isolate the other component as well: $(\mathbb{1}, (n-1-k, n-k))s_a$ is either $((0, \dots, n-2-k), \mathbb{1})$ or $((1, \dots, n-2-k), \mathbb{1})$. Paired with $((0, 1), \mathbb{1})$, either element is sufficient to generate $S_{n-1-k} \times \{\mathbb{1}\}$. Similarly, s_a and s_b generate $\{\mathbb{1}\} \times S_k$. Therefore s_a and s_b generate $S_{n-1-k} \times S_k$. It follows that a and b generate all permutations t of Q_n such that $E_{n,k}t = E_{n,k}$, $F_{n,k}t = F_{n,k}$, and $(n-1)t = n-1$. □

Proposition 3 (Syntactic Semigroup). *The syntactic semigroup of $L_{n,k}(\Sigma)$ has cardinality $n^{n-1-k}(k+1)^k$, which is maximal for a k-proper language. Furthermore, seven letters are required to meet this bound. The maximum value $n(n-1)^{n-2}$ is reached only when $k = n-2$.*

Proof. Let L be a k-proper language of complexity n and let $\mathcal{D}$ be a minimal DFA recognizing L. By Lemma 1, $\mathcal{D}$ has an empty state. By Proposition 1, the only states that can be reached from one of the k final states are either final or empty. Thus, a transformation in the transition semigroup of $\mathcal{D}$ may map each final state to one of $k+1$ possible states, while each non-final, non-empty state may be mapped to any of the n states. Since the empty state can only be mapped to itself, we are left with $n^{n-1-k}(k+1)^k$ possible transformations in the transition semigroup. Therefore the syntactic semigroup of any k-proper language has size at most $n^{n-1-k}(k+1)^k$.

Now consider the transition semigroup of $\mathcal{D}_{n,k}(\Sigma)$. Every transformation t in the semigroup must satisfy $F_{n,k}t \subseteq F_{n,k} \cup \{n-1\}$ and $(n-1)t = n-1$, since any other transformation would violate prefix-convexity. We show that the semigroup contains every such transformation, and hence the syntactic semigroup of $L_{n,k}(\Sigma)$ is maximal.

First, consider the transformations t such that $E_{n,k}t \subseteq E_{n,k} \cup \{n-1\}$ and $qt = q$ for all $q \in F_{n,k} \cup \{n-1\}$. By Lemma 2, a and b generate every permutation of $E_{n,k}$. When t is not a permutation, we can use c_1 to combine any states p and q: apply a permutation on $E_{n,k}$ so that $p \to 0$ and $q \to 1$, and then apply c_1 so that $1 \to 0$. Repeat this method to combine any set of states, and further apply permutations to induce the desired transformation while leaving the states of $F_{n,k} \cup \{n-1\}$ in place. The same idea applies with d_1; apply permutations and d_1 to send any states of $E_{n,k}$ to $n-1$. Hence a, b, c_1, and d_1 generate every transformation t such that $E_{n,k}t \subseteq E_{n,k} \cup \{n-1\}$ and $qt = q$ for all $q \in F_{n,k} \cup \{n-1\}$.

We can make the same argument for transformations that act only on $F_{n,k}$ and fix every other state. Since c_2 and d_2 act on $F_{n,k}$ exactly as c_1 and d_1 act on $E_{n,k}$, the letters a, b, c_2, and d_2 generate every transformation t such that $F_{n,k}t \subseteq F_{n,k} \cup \{n-1\}$ and $qt = q$ for all $q \in E_{n,k} \cup \{n-1\}$. It follows that a, b, c_1, c_2, d_1, and d_2 generate every transformation t such that $E_{n,k}t \subseteq E_{n,k} \cup \{n-1\}$, $F_{n,k}t \subseteq F_{n,k} \cup \{n-1\}$, and $(n-1)t = n-1$.

Note the similarity between this DFA restricted to the states $E_{n,k} \cup \{n-1\}$ (or $F_{n,k} \cup \{n-1\}$) and the witness for right ideals introduced in [7]. The argument for the size of the syntactic semigroup of right ideals is similar to this; see [10].

Finally, consider an arbitrary transformation t such that $F_{n,k}t \subseteq F_{n,k} \cup \{n-1\}$ and $(n-1)t = n-1$. Let j_t be the number of states $p \in E_{n,k}$ such that $pt \in F_{n,k}$. We show by induction on j_t that t is in the transition semigroup of $\mathcal{D}$. If $j_t = 0$, then t is generated by $\Sigma \setminus \{e\}$. If $j_t \geqslant 1$, there exist $p, q \in E_{n,k}$ such that $pt \in F_{n,k}$ and q is not in the image of t. Consider the transformations s_1 and s_2 defined by $qs_1 = pt$ and $rs_1 = r$ for $r \neq q$, and $ps_2 = q$ and $rs_2 = rt$ for $r \neq p$. Then $(rs_2)s_1 = rt$ for all $r \in Q_n$. Notice that $j_{s_2} = j_t - 1$, and hence Σ generates s_2 by inductive assumption. One can verify that $s_1 = (n-1-k, pt)(0, q)(0 \to n-1-k)(0, q)(n-1-k, pt)$. From this expression, we see that s_1 is the composition of transpositions induced by words in $\{a, b\}^*$ and the transformation $(0 \to n-1-k)$ induced by e, and hence s_1 is generated by Σ. Thus, t is in the transition semigroup. By induction on j_t, it follows that the syntactic semigroup of $L_{n,k}$ is maximal.

Now we show that seven letters are required to meet this bound. Two letters (like a and b) are required to generate the permutations, since clearly one letter is not sufficient. Every other letter will induce a properly injective map. A letter (like c_1) that induces a properly injective map on $E_{n,k}$ and permutes $F_{n,k}$ is required. Similarly, a letter (like c_2) that permutes $E_{n,k}$ and induces a properly injective map on $F_{n,k}$ is required. A letter (like d_1) that sends a state in $E_{n,k}$ to $n-1$ and permutes $F_{n,k}$ is required. Similarly, a letter (like d_2) that sends a state in $F_{n,k}$ to $n-1$ and permutes $E_{n,k}$ is required. Finally, a letter (like e) that connects $E_{n,k}$ and $F_{n,k}$ is required.

For a fixed n, we may want to know which $k \in \{1, \ldots, n-2\}$ maximizes $s_n(k) = n^{n-1-k}(k+1)^k$; this corresponds to the largest syntactic semigroup of a proper prefix-convex language with n quotients. We show that $s_n(k)$ is largest at $k = n-2$. Consider the ratio $\frac{s_n(k+1)}{s_n(k)} = \frac{(k+2)^{k+1}}{n(k+1)^k}$. Notice this ratio is increasing with k, and hence s_n is a convex function on $\{1, \ldots, n-2\}$. It follows that the maximum value of s_n must occur at one the endpoints, 1 and $n-2$.

Now we show that $s_n(n-2) \geqslant s_n(1)$ for all $n \geqslant 3$. We can check this explicitly for $n = 3, 4, 5$. When $n \geqslant 6$, $s_n(n-2)/s_n(1) = \frac{n}{2}\left(\frac{n-1}{n}\right)^{n-2} \geqslant 3\,(1/e) > 1$; so the largest syntactic semigroup of $L_{n,k}(\Sigma)$ occurs only at $k = n-2$ for all $n \geqslant 3$. □

Proposition 4 (Reverse). *For any regular language L of complexity n with an empty quotient, the reversal has complexity at most 2^{n-1}. Moreover, the reverse of $L_{n,k}(a,b,-,-,-,d_2,e)$ has complexity 2^{n-1} for $n \geqslant 3$ and $1 \leqslant k \leqslant n-2$.*

Proof. The first claim is left for the reader to verify. For the second claim, let $\mathcal{D}_{n,k} = (Q_n, \{a,b,d_2,e\}, \delta_{n,k}, 0, F_{n,k})$ denote the DFA $\mathcal{D}_{n,k}(a,b,-,-,-,d_2,e)$ in Definition 2 and let $L_{n,k} = L(D_{n,k})$. Construct an NFA $\mathcal{N}$ recognizing the reverse of $L_{n,k}$ by reversing each transition, letting the initial state 0 be the unique final state, and letting the final states in $F_{n,k}$ be the initial states. Applying the subset construction to $\mathcal{N}$ yields a DFA $\mathcal{D}^R$ whose states are subsets of Q_{n-1}, with initial state $F_{n,k}$ and final states $\{U \subseteq Q_{n-1} \mid 0 \in U\}$. We show that $\mathcal{D}^R$ is minimal, and hence the reverse of $L_{n,k}$ has complexity 2^{n-1}.

Recall from Lemma 2 that a and b generate all permutations of $E_{n,k}$ and $F_{n,k}$ in $\mathcal{D}_{n,k}$ and, although the transitions are reversed in $\mathcal{D}^R$, they still generate all such permutations. Let $u_1, u_2 \in \{a,b\}^*$ be such that u_1 induces $(0,\dots,n-2-k)$ and u_2 induces $(n-1-k,\dots,n-2)$ in $\mathcal{D}^R$.

Consider a state $U = \{q_1,\dots,q_h, n-1-k,\dots,n-2\}$ where $0 \leqslant q_1 < q_2 < \cdots < q_h \leqslant n-2-k$. If $h=0$, then U is the initial state. When $h \geqslant 1$, $\{q_2-q_1, q_3-q_1,\dots,q_h-q_1, n-1-k,\dots,n-2\}eu_1^{q_1} = U$. By induction, all such states are reachable.

Now we show that any state $U = \{q_1,\dots,q_h,p_1,\dots,p_i\}$ where $0 \leqslant q_1 < q_2 < \cdots < q_h \leqslant n-2-k$ and $n-1-k \leqslant p_1 < p_2 < \cdots < p_i \leqslant n-2$ is reachable. If $i=k$, then $U = \{q_1,\dots,q_h,n-1-k,\dots,n-2\}$ is reachable by the argument above. When $0 \leqslant i < k$, choose $p \in F_{n,k} \setminus U$ and see that U is reached from $U \cup \{p\}$ by $u_2^{n-1-p} d_2 u_2^{p-(n-2-k)}$. By induction, every state is reachable.

To prove distinguishability, consider distinct states U and V. Choose $q \in U \oplus V$. If $q \in E_{n,k}$, then U and V are distinguished by $u_1^{n-1-k-q}$. When $q \in F_{n,k}$, they are distinguished by $u_2^{n-1-q}e$. So $\mathcal{D}^R$ is minimal. □

Proposition 5 (Star). *Let L be a regular language with $n \geqslant 2$ quotients, including $k \geqslant 1$ final quotients and one empty quotient. Then $\kappa(L^*) \leqslant 2^{n-2} + 2^{n-2-k}+1$. This bound is tight for prefix-convex languages; in particular, the language $(L_{n,k}(a,b,-,-,d_1,d_2,e))^*$ meets this bound for $n \geqslant 3$ and $1 \leqslant k \leqslant n-2$.*

Proof. Since L has an empty quotient, let $n-1$ be the empty state of its minimal DFA $\mathcal{D}$. To obtain an ε-NFA for L^*, we add a new initial state $0'$ which is final and has the same transitions as 0. We then add an ε-transition from every state in F to 0. Applying the subset construction to this ε-NFA yields a DFA $\mathcal{D}' = (Q', \Sigma, \delta', \{0'\}, F')$ recognizing L^*, in which Q' contains non-empty subsets of $Q_n \cup \{0'\}$.

Many of the states of Q' are unreachable or indistinguishable from other states. Since there is no transition in the ε-NFA to $0'$, the only reachable state in Q' containing $0'$ is $\{0'\}$. As well, any reachable final state $U \neq \{0'\}$ must contain 0 because of the ε-transitions. Finally, for any $U \in Q'$, we have $U \in F'$ if and only if $U \cup \{n-1\} \in F'$, and since $\delta'(U \cup \{n-1\}, w) = \delta'(U,w) \cup \{n-1\}$ for all $w \in \Sigma^*$, the states U and $U \cup \{n-1\}$ are equivalent in D'.

Hence $\mathcal{D}'$ is equivalent to a DFA with the states $\{\{0'\}\} \cup \{U \subseteq Q_{n-1} \mid U \cap F = \emptyset\} \cup \{U \subseteq Q_{n-1} \mid 0 \in U \text{ and } U \cap F \neq \emptyset\}$. This DFA has $1 + 2^{n-1-k} + (2^{n-2} - 2^{n-2-k}) = 2^{n-2} + 2^{n-2-k} + 1$ states. Thus, $\kappa(L^*) \leqslant 2^{n-2} + 2^{n-2-k} + 1$.

This bound applies when L is a prefix-convex language and $n \geqslant 3$. By Lemma 1, L is either a right ideal or has an empty state. If L is a right ideal, then $\kappa(L^*) \leqslant n+1$, which is at most $2^{n-2} + 2^{n-2-k} + 1$ for $n \geqslant 3$.

For the last claim, let $\mathcal{D}_{n,k}(a, b, -, -, d_1, d_2, e)$ of Definition 2 be denoted by $\mathcal{D}_{n,k} = (Q_n, \{a, b, d_1, d_2, e\}, \delta_{n,k}, 0, F_{n,k})$ and let $L_{n,k} = L(D_{n,k})$. We apply the same construction and reduction as before to obtain a DFA $\mathcal{D}'_{n,k}$ recognizing $L^*_{n,k}$ with states $Q' = \{\{0'\}\} \cup \{U \subseteq E_{n,k}\} \cup \{U \subseteq Q_{n-1} \mid 0 \in U \text{ and } U \cap F_{n,k} \neq \emptyset\}$. We show that the states of Q' are reachable and pairwise distinguishable.

By Lemma 2, a and b generate all permutations of $E_{n,k}$ and $F_{n,k}$ in $\mathcal{D}_{n,k}$. Choose $u_1, u_2 \in \{a, b\}^*$ such that u_1 induces $(0, \ldots, n-2-k)$ and u_2 induces $(n-1-k, \ldots, n-2)$ in $\mathcal{D}_{n,k}$.

For reachability, we consider three cases. (1) State $\{0'\}$ is reachable by ε. (2) Let $U \subseteq E_{n,k}$. For any $q \in E_{n,k}$, we can reach $U \setminus \{q\}$ by $u_1^{n-2-k-q} d_1 u_1^q$; hence if U is reachable, then every subset of U is reachable. Observe that state $E_{n,k}$ is reachable by $e u_1^{n-2-k} d_2^k$, and we can reach any subset of this state. Therefore, all non-final states are reachable. (3) If $U \cap F_{n,k} \neq \emptyset$, then $U = \{0, q_1, q_2, \ldots, q_h, r_1, \ldots, r_i\}$ where $0 < q_1 < \cdots < q_h \leqslant n-2-k$ and $n-1-k \leqslant r_1 < \cdots < r_i < n-1$ and $i \geqslant 1$. We prove that U is reachable by induction on i. If $i = 0$, then U is reachable by (2). For any $i \geqslant 1$, we can reach U from $\{0, q_1, \ldots, q_h, r_2 - (r_1 - (n-1-k)), \ldots, r_i - (r_1 - (n-1-k))\}$ by $e u_2^{r_1-(n-1-k)}$. Therefore, all states of this form are reachable.

Now we show that the states are pairwise distinguishable. (1) The initial state $\{0'\}$ is distinguishable from any other final state U since $\{0'\}u_1$ is non-final and Uu_1 is final. (2) If U and V are distinct subsets of $E_{n,k}$, then there is some $q \in U \oplus V$. We distinguish U and V by $u_1^{n-1-k-q} e$. (3) If U and V are distinct and final and neither one is $\{0'\}$, then there is some $q \in U \oplus V$. If $q \in E_{n,k}$, then $U d_2^k = U \setminus F_{n,k}$ and $V d_2^k = V \setminus F_{n,k}$ are distinct, non-final states as in (2). Otherwise, $q \in F_{n,k}$ and we distinguish U and V by $u_2^{n-1-q} d_2^{k-1}$. □

Table 1. Complexities of prefix-convex languages

	Right-ideal	Prefix-closed	Prefix-free	Proper
SeGr	$\mathbf{n^{n-1}}$	$\mathbf{n^{n-1}}$	n^{n-2}	$n^{n-1-k}(k+1)^k$
Rev	$\mathbf{2^{n-1}}$	$\mathbf{2^{n-1}}$	$2^{n-2}+1$	$\mathbf{2^{n-1}}$
Star	$n+1$	$2^{n-2}+1$	n	$\mathbf{2^{n-2}+2^{n-2-k}+1}$
Prod	$m+2^{n-2}$	$(m+1)2^{n-2}$	$m+n-2$	$\mathbf{m-1-j+j2^{n-2}+2^{n-1}}$
$\cup$	$mn-(m+n-2)$	$\mathbf{mn}$	$mn-2$	$\mathbf{mn}$
$\oplus$	$\mathbf{mn}$	$\mathbf{mn}$	$mn-2$	$\mathbf{mn}$
$\setminus$	$\mathbf{mn-(m-1)}$	$\mathbf{mn-(n-1)}$	$mn-(m+2n-4)$	$\mathbf{mn-(n-1)}$
$\cap$	$\mathbf{mn}$	$mn-(m+n-2)$	$mn-2(m+n-3)$	$mn-(m+n-2)$

3 Conclusions

The bounds for prefix-convex languages (see also [8]) are summarized in Table 1. The largest bounds are shown in boldface type, and they are reached either in the class of right-ideal languages or the class of proper languages. Recall that for regular languages we have the following results: semigroup n^n, reverse 2^n, star $2^{n-1} + 2^{n-2}$, product $m2^n - 2^{n-1}$, boolean operations mn.

References

1. Ang, T., Brzozowski, J.A.: Languages convex with respect to binary relations, and their closure properties. Acta Cybernet. **19**(2), 445–464 (2009)
2. Berstel, J., Perrin, D., Reutenauer, C.: Codes and Automata (Encyclopedia of Mathematics and its Applications). Cambridge University Press, New York (2010)
3. Brzozowski, J.A.: Quotient complexity of regular languages. J. Autom. Lang. Comb. **15**(1/2), 71–89 (2010)
4. Brzozowski, J.A.: In search of the most complex regular languages. Int. J. Found. Comput. Sci **24**(6), 691–708 (2013)
5. Brzozowski, J.A., Davies, S., Liu, B.Y.V.: Most complex regular ideal languages. Discrete Math. Theoret. Comput. Sci. **18**(3), 1–25 (2016). Paper #15
6. Brzozowski, J.A., Jirásková, G., Zou, C.: Quotient complexity of closed languages. Theory Comput. Syst. **54**, 277–292 (2014)
7. Brzozowski, J.A., Sinnamon, C.: Complexity of prefix-convex regular languages (2016). http://arxiv.org/abs/1605.06697
8. Brzozowski, J.A., Sinnamon, C.: Complexity of right-ideal, prefix-closed, and prefix-free regular languages. Acta Cybernet. (2017, to appear)
9. Brzozowski, J.A., Tamm, H.: Theory of átomata. Theoret. Comput. Sci. **539**, 13–27 (2014)
10. Brzozowski, J., Ye, Y.: Syntactic complexity of ideal and closed languages. In: Mauri, G., Leporati, A. (eds.) DLT 2011. LNCS, vol. 6795, pp. 117–128. Springer, Heidelberg (2011). doi:10.1007/978-3-642-22321-1_11
11. Crochemore, M., Hancart, C.: Automata for pattern matching. In: Rozenberg, G., Salomaa, A. (eds.) Handbook of Formal Languages, vol. 2, pp. 399–462. Springer, Heidelberg (1997)
12. Holzer, M., König, B.: On deterministic finite automata and syntactic monoid size. Theoret. Comput. Sci. **327**(3), 319–347 (2004)
13. Iván, S.: Complexity of atoms, combinatorially. Inform. Process. Lett. **116**(5), 356–360 (2016)
14. 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.) CIAA 2004. LNCS, vol. 3317, pp. 213–224. Springer, Heidelberg (2005). doi:10.1007/978-3-540-30500-2_20
15. Maslov, A.N.: Estimates of the number of states of finite automata. Dokl. Akad. Nauk SSSR **194**, 1266–1268 (1970). (Russian). English translation: Soviet Math. Dokl. **11**, 1373–1375 (1970)
16. Thierrin, G.: Convex languages. In: Nivat, M. (ed.) Automata, Languages and Programming, pp. 481–492. North-Holland (1973)
17. Yu, S.: State complexity of regular languages. J. Autom. Lang. Comb. **6**, 221–234 (2001)
18. Yu, S., Zhuang, Q., Salomaa, K.: The state complexities of some basic operations on regular languages. Theoret. Comput. Sci. **125**(2), 315–328 (1994)