View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by University of Waterloo's Institutional Repository

Journal of Automata, Languages and Combinatorics **u** (v) w, x–y
© Otto-von-Guericke-Universität Magdeburg

Syntactic Complexities of Six Classes of Star-Free Languages [1]

Janusz Brzozowski, Baiyu Li [2]
David R. Cheriton School of Computer Science
University of Waterloo, Waterloo, ON, Canada N2L 3G1
e-mail: {brzozo,b5li}@uwaterloo.ca

and

David Liu [3]
Department of Computer Science
University of Toronto Toronto, ON, Canada M5S 3G4
e-mail: liudavid@cs.toronto.edu

ABSTRACT

The syntactic complexity of a regular language is the cardinality of its syntactic semigroup. The syntactic complexity of a subclass of regular languages is the maximal syntactic complexity of languages in that subclass, taken as a function of the state complexity n of these languages. We study the syntactic complexity of six subclasses of star-free languages. We find a tight upper bound of $(n-1)!$ for finite/cofinite and reverse definite languages, and a lower bound of $\lfloor \mathbf{e}\cdot(n-1)! \rfloor$ for definite languages, where $\mathbf{e}$ is the base of the natural logarithms. We also find tight upper bounds for languages accepted by monotonic, partially monotonic and "nearly monotonic" automata. All these bounds are significantly lower than the bound n^n for arbitrary regular languages. Also, witness languages reaching these bounds require alphabets that grow with n. The syntactic complexity of arbitrary star-free languages remains open.

Keywords: cofinite language, definite language, finite automaton, finite language, monotonic automaton, partially monotonic automaton, reverse definite language, star-free language, syntactic complexity, syntactic semigroup

1. Introduction

Star-free languages are the smallest class containing the finite languages and closed under boolean operations and concatenation. In 1965, Schützenberger [26] proved that a language is star-free if and only if its syntactic monoid is *group-free*, that is, has only trivial subgroups. An equivalent condition is that in the minimal deterministic finite automaton (DFA) of a star-free language no word can induce a permutation of

[1] This work was supported by the Natural Sciences and Engineering Research Council of Canada under grant No. OGP0000871.
[2] Present address: Optumsoft, Inc., 275 Middlefield Rd, Suite 210, Menlo Park, CA 94025, USA.
[3] This work was done while David Liu was at the University of Waterloo.

any set of two or more states, other than the identity permutation. Such automata are called *aperiodic*; they were studied in 1971 by McNaughton and Papert [21].

The *state complexity of a regular language* is the number of states in the minimal DFA recognizing that language. State complexity of operations on languages has been studied quite extensively; for a survey of this topic and a list of references see [28]. An equivalent notion is that of *quotient complexity* [4], which is the number of left quotients of the language.

Quotient complexity is closely related to the Nerode equivalence [23]. Another well-known equivalence relation, the Myhill equivalence [22], defines the syntactic semigroup of a language and its *syntactic complexity*, which is the cardinality of the syntactic semigroup. It was pointed out in [8] that syntactic complexity can be very different for languages with the same quotient complexity.

In contrast to state complexity, syntactic complexity has not received much attention until recently. Suppose L is a regular language with quotient complexity n. In 1970 Maslov [20] noted that n^n is a tight upper bound on the syntactic complexity of L. In 2003–2004 Holzer and König [15], and Krawetz, Lawrence and Shallit [18] studied the syntactic complexity of unary and binary languages. In 2011 Brzozowski and Ye [8] showed that, if L is any right ideal, then n^{n-1} is a tight upper bound on its syntactic complexity. They also proved that $n^{n-1} + (n-1)$ (respectively, $n^{n-2} + (n-2)2^{n-2} + 1$) is a lower bound if L is a left (respectively, two-sided) ideal. In 2012 Brzozowski, Li and Ye [6] showed that n^{n-2} is a tight upper bound for prefix-free languages and that $(n-1)^{n-2} + (n-2)$ (respectively, $(n-1)^{n-3} + (n-2)^{n-3} + (n-3)2^{n-3}$ or $(n-1)^{n-3} + (n-3)2^{n-3} + 1$) is a lower bound for suffix-free (respectively, bifix-free or factor-free) languages.

Here we deal with the syntactic complexity of six families of star-free languages. We start with the simplest family, that of finite and cofinite languages. Testing whether a word belongs to a finite or cofinite language can be done by checking a list of words shorter than some fixed length. We also study definite and reverse definite languages.

A language is *definite* if it can be decided whether a word w belongs to it simply by examining the suffix of w of some fixed length. The class of definite languages was the very first subclass of regular languages to be considered: it was introduced in 1954 in the classic paper by Kleene [17]. It was then studied in 1963 by Perles, Rabin, and Shamir [24], and Brzozowski [3], in 1966 by Ginzburg [12], and later by several others. Definite languages were revisited in 2009 by Bordihn, Holzer and Kutrib [2] in connection with state complexity.

Reverse definite languages were first studied by Brzozowski [3]. Here membership of w can be determined by its prefix of some fixed length. The class of *finite and cofinite* languages is the intersection of the class of definite languages with the class of reverse definite languages.

All three classes, finite/cofinite, definite and reverse definite, are boolean algebras. The following characterizations of finite/cofinite, definite and reverse definite subsets of Σ^+ in terms their syntactic semigroups can be found in Eilenberg's book, volume B [10]. The semigroup S of a finite/cofinite language is *nilpotent*: It is characterized by the condition $iS = Si = i$ for every idempotent i in S. Equivalently, either S is empty or has a zero and no other idempotent. For definite (respectively, reverse definite) languages every idempotent i is a right zero, that is, $Si = i$ (respectively, a left zero, that is, $iS = i$).

We discovered that the syntactic complexity problem gets difficult very quickly: Although finite/cofinite and reverse definite languages were relatively easy, we have been unable to establish a tight upper bound for definite languages.

Using known results from semigroup theory, we found the complexity $\binom{2n-1}{n}$ of languages accepted by monotonic automata, and discovered that partially monotonic automata lead to much higher complexities. We then found even higher complexities with "nearly monotonic" automata.

Two other subclasses of star-free languages were studied in [5]. The syntactic complexity of $\mathcal{R}$-trivial languages is $n!$, and of $\mathcal{J}$-trivial languages, it is $\lfloor \mathbf{e} \cdot (n-1)! \rfloor$; those results are beyond the scope of this paper.

For large values of n, the list of subclasses with syntactic complexities in increasing order is as follows: 1) monotonic, 2) partially monotonic, 3) nearly monotonic, 4) finite/cofinite and reverse definite, 5) $\mathcal{J}$-trivial, 6) definite ? (upper bound not known), 7) $\mathcal{R}$-trivial. Recently, Marek Szykuła[4] found subclasses of star-free languages with even larger semigroups.

Our terminology and some basic facts are stated in Section 2. Aperiodic transformations are examined in Section 3. In Section 4, we study finite/cofinite, reverse definite, and definite languages. In Section 5, we study monotonic, partially monotonic, and nearly monotonic automata and languages. Section 6 concludes the paper.

2. Preliminaries

We assume the reader is familiar with basic theory of formal languages as in [25] for example. Let Σ be a non-empty finite alphabet and Σ^*, the free monoid generated by Σ. A *word* is any element of Σ^*, and the empty word is ε. The length of a word $w \in \Sigma^*$ is $|w|$. A *language* over Σ is any subset of Σ^*. For any languages K and L over Σ, we use the *boolean operations*: complement ($\overline{L}$) and union ($K \cup L$). The *product*, or *(con)catenation*, of K and L is $KL = \{w \in \Sigma^* \mid w = uv, u \in K, v \in L\}$; the *star* of L is $L^* = \bigcup_{i \geqslant 0} L^i$, and the *positive closure* of L is $L^+ = \bigcup_{i \geqslant 1} L^i$.

We call languages $\emptyset$, $\{\varepsilon\}$, and $\{a\}$ for any $a \in \Sigma$ the *basic* languages. *Regular* languages are the class of languages constructed from the basic languages using only boolean operations, product, and star. *Star-free* languages are the class of languages constructed from the basic languages using only boolean operations and product.

A *deterministic finite automaton* (DFA) is a quintuple $\mathcal{D} = (Q, \Sigma, \delta, q_1, F)$, where Q is a finite, non-empty set of *states*, Σ is a finite *alphabet*, $\delta \colon Q \times \Sigma \to Q$ is the *transition function*, $q_1 \in Q$ is the *initial state*, and $F \subseteq Q$ is the set of *final states*. We extend δ to $Q \times \Sigma^*$ in the usual way. The DFA $\mathcal{D}$ accepts a word $w \in \Sigma^*$ if $\delta(q_1, w) \in F$. The set of all words *accepted* by $\mathcal{D}$ is $L(\mathcal{D})$. Regular languages are exactly the languages accepted by DFAs. By the *language of a state* q of $\mathcal{D}$ we mean the language $L_q(\mathcal{D})$ accepted by the DFA $(Q, \Sigma, \delta, q, F)$. A state is *empty* (also called *dead* or a *sink* state) if its language is empty. Two states p and q of $\mathcal{D}$ are *equivalent* if

[4] personal communication

$L_p(\mathcal{D}) = L_q(\mathcal{D})$. Otherwise, there exists a word $w \in L_p(\mathcal{D}) \oplus L_q(\mathcal{D})$, where $\oplus$ denotes symmetric difference, and states p and q are *distinguishable*. A DFA is *minimal* if all states are reachable and pairwise distinguishable.

An *incomplete deterministic finite automaton (IDFA)* is a quintuple $\mathcal{I} = (Q, \Sigma, \delta, q_1, F)$, where Q, Σ, q_1 and F are as in a DFA, and δ is a partial function. Every DFA is also an IDFA.

The *left quotient*, or simply *quotient,* of a language L by a word w is the language $w^{-1}L = \{x \in \Sigma^* \mid wx \in L\}$. The *Nerode equivalence* $\sim_L$ of any language L over Σ is defined as follows [23]: For all $x, y \in \Sigma^*$,

$$x \sim_L y \text{ if and only if } xv \in L \Leftrightarrow yv \in L, \text{ for all } v \in \Sigma^*.$$

Clearly, $x^{-1}L = y^{-1}L$ if and only if $x \sim_L y$. Thus each equivalence class of the Nerode equivalence corresponds to a distinct quotient of L.

Let L be a regular language. The *quotient DFA* of L is $\mathcal{D} = (Q, \Sigma, \delta, q_1, F)$, where $Q = \{w^{-1}L \mid w \in \Sigma^*\}$, $\delta(w^{-1}L, a) = (wa)^{-1}L$, $q_1 = \varepsilon^{-1}L = L$, and $F = \{w^{-1}L \mid \varepsilon \in w^{-1}L\}$. Every quotient DFA is minimal. The *quotient IDFA* of L is the quotient DFA of L after the empty state, if present, and all transitions incident to it are removed. The quotient IDFA is also minimal. If a regular language L has quotient IDFA $\mathcal{I}$, then the DFA $\mathcal{D}$ obtained by adding the empty state to $\mathcal{I}$, if necessary, is the quotient DFA of L. The two automata $\mathcal{D}$ and $\mathcal{I}$ accept the same language.

The number $\kappa(L)$ of distinct quotients of L is the *quotient complexity* of L. Since the quotient DFA of L is minimal, quotient complexity is the same as state complexity.

The *Myhill equivalence* $\approx_L$ of L is defined as follows [22]: For all $x, y \in \Sigma^*$,

$$x \approx_L y \text{ if and only if } uxv \in L \Leftrightarrow uyv \in L \text{ for all } u, v \in \Sigma^*.$$

This equivalence is also known as the *syntactic congruence* of L, and the quotient $\Sigma^*/{\approx_L}$ is the *syntactic monoid* of L.

We also use $\approx_L$ restricted to Σ^+, that is, define $\approx_L$ only for $x, y \in \Sigma^+$; then $\Sigma^+/{\approx_L}$ is the *syntactic semigroup* of L, which we denote by S_L. The *syntactic complexity* $\sigma(L)$ of L is the cardinality of its syntactic semigroup.

Although it would be possible to state our results entirely in terms of subsets of Σ^+, there are some serious disadvantages in doing so. One would then have to use complementation with respect to Σ^+. The DFA for the complement $\overline{L}$ of a language L would no longer be obtained from the DFA of L by interchanging the sets of final and non-final states. The state complexity of $\overline{L}$ would no longer be the same as that of L. Also, almost all literature on regular languages deals with subsets of Σ^*. For these reasons, we deal with languages as subsets of Σ^* as usual, but measure their syntactic complexity by the size of the syntactic semigroup.

In Section 1, we mentioned the characterizations of finite/cofinite, definite, and reverse definite languages in terms of syntactic semigroups. Such results cannot be stated in terms of syntactic monoids, since the identity transformation induced by the empty word is an idempotent that does not satisfy the stated conditions. For consistency, we also use syntactic semigroups with monotonic, partially monotonic, and nearly monotonic languages studied in Section 5.

A *partial transformation* of a set Q is a partial mapping of Q into itself; we consider partial transformations of finite sets only, and assume without loss of generality that $Q = \{1, 2, \ldots, n\}$. Let t be a partial transformation of Q. If t is defined for $i \in Q$, then it is the image of i under t; otherwise it is undefined and we write $it = \Box$. For convenience, we let $\Box t = \Box$. If X is a subset of Q, then $Xt = \{it \mid i \in X\}$. The *composition* of two partial transformations t_1 and t_2 of Q is a partial transformation $t_1 \circ t_2$ such that $i(t_1 \circ t_2) = (it_1)t_2$ for all $i \in Q$. We usually drop the composition operator "$\circ$" and write t_1t_2 for short.

An arbitrary partial transformation can be written in the form

$$t = \begin{pmatrix} 1 & 2 & \cdots & n-1 & n \\ i_1 & i_2 & \cdots & i_{n-1} & i_n \end{pmatrix},$$

where $i_k = kt$ and $i_k \in Q \cup \{\Box\}$, for $k \in Q$. The *domain* of t is the set $\mathrm{dom}(t) = \{k \in Q \mid kt \neq \Box\}$. The *range* of t is the set $\mathrm{rng}(t) = \mathrm{dom}(t)t = \{kt \mid k \in Q \text{ and } kt \neq \Box\}$. When the domain is clear, we also write $t = [i_1, \ldots, i_n]$.

A *(full) transformation* t of Q is a partial transformation such that $\mathrm{dom}(t) = Q$. Let $\mathcal{T}_Q$ be the set of all transformations of Q; then $\mathcal{T}_Q$ is a semigroup under composition. The *identity* transformation $\mathbf{1}$ maps each element to itself, that is, $i\mathbf{1} = i$ for all $i \in Q$. A transformation t is a *cycle* of length $k \geqslant 2$ if there exist pairwise distinct elements $i_1, \ldots, i_k$ such that $i_1t = i_2, i_2t = i_3, \ldots, i_{k-1}t = i_k$, $i_kt = i_1$, and $jt = j$ for all $j \notin \{i_1, \ldots, i_k\}$. Such a cycle is denoted by $(i_1, i_2, \ldots, i_k)$. For $i < j$, a *transposition* is the cycle (i, j). A *singular* transformation, denoted by $(i \to j)$, has $it = j$ and $ht = h$ for all $h \neq i$. A *constant* transformation, denoted by $(Q \to j)$, has $it = j$ for all i. A *permutation* of Q is a mapping of Q *onto* itself. A transformation t is *permutational* if there exists some $X \subseteq Q$ with $|X| \geqslant 2$ such that $t|_X$ is a permutation of X. Otherwise, t is *non-permutational*.

Let $\mathcal{D} = (Q, \Sigma, \delta, q_1, F)$ be a DFA. For each word $w \in \Sigma^+$, the transition function defines a transformation t_w of Q: for all $i \in Q$, $it_w \stackrel{\text{def}}{=} \delta(i, w)$. The set $T_{\mathcal{D}}$ of all such transformations by non-empty words forms a subsemigroup of $\mathcal{T}_Q$, called the *transition semigroup* of $\mathcal{D}$ [25]. Conversely, we can use a set $\{t_a \mid a \in \Sigma\}$ of transformations to define δ, and so the DFA $\mathcal{D}$. When the context is clear we write $a\colon t$, where t is a transformation of Q, to mean that the transformation performed by $a \in \Sigma$ is t. If $\mathcal{D}$ is the minimal DFA of L, then $T_{\mathcal{D}}$ is isomorphic to the syntactic semigroup S_L of L [21], and we represent elements of S_L by transformations in $T_{\mathcal{D}}$.

For any IDFA $\mathcal{I}$, each word $w \in \Sigma^*$ performs a partial transformation of Q. The set of all such partial transformations is the *transition semigroup* of $\mathcal{I}$. If $\mathcal{I}$ is the minimal IDFA of a language L, this semigroup is isomorphic to the transition semigroup of the minimal DFA of L, and hence also to the syntactic semigroup of L.

3. Aperiodic Transformations

A transformation is *aperiodic* if it contains no cycles of length greater than 1. A semigroup T of transformations is *aperiodic* if and only if it contains only aperiodic trans-

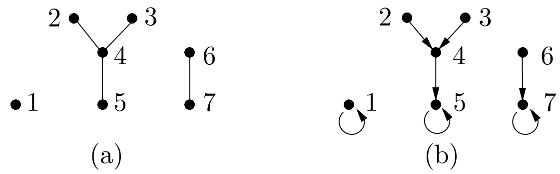

Figure 1: Forests and aperiodic transformations.

formations. Thus a language L with minimal DFA $\mathcal{D}$ is star-free if and only if every transformation in $S_{\mathcal{D}}$ is aperiodic.

Let A_n be the set of all aperiodic transformations of Q. Each aperiodic transformation can be characterized by a forest of labeled rooted trees as follows. Consider, for example, the forest of Figure 1 (a), where the roots are at the bottom. Convert this forest into a directed graph by adding a direction from each child to its parent and a self-loop to each root, as shown in Figure 1 (b). This directed graph defines the transformation $[1, 4, 4, 5, 5, 7, 7]$ and such a transformation is aperiodic since the directed graph has no cycles of length greater than one. Thus there is a one-to-one correspondence between aperiodic transformations of a set of n elements and forests with n nodes.

Proposition 1 *There are $(n+1)^{n-1}$ aperiodic transformations of a set Q of $n \geqslant 1$ elements.*

Proof. By Cayley's theorem [9, 27], there are $(n+1)^{n-1}$ labeled unrooted trees with $n+1$ nodes. If we fix one node, say node $n+1$, in each of these trees to be the root, then we have $(n+1)^{n-1}$ labeled trees rooted at $n+1$. Let T be any one of these trees, and let $v_1, \dots, v_m$ be the children of $n+1$ in T. By removing the root $n+1$ from T, we get a labeled forest F with n nodes formed by m rooted trees, where $v_1, \dots, v_m$ are the roots. Then we get an aperiodic transformation of $\{1, \dots, n\}$ by adding self-loops on $v_1, \dots, v_m$.

All labeled rooted forests with n nodes can be obtained uniquely from some rooted tree with $n+1$ nodes by deleting the root. Hence there are $(n+1)^{n-1}$ labeled rooted forests with n nodes, and that many aperiodic transformations of Q. □

Since the minimal DFA of a star-free language can perform only aperiodic transformations, we have

Corollary 2 *For $n \geqslant 1$, the syntactic complexity $\sigma(L)$ of a star-free language L with n quotients satisfies $\sigma(L) \leqslant (n+1)^{n-1}$.*

The bound of Corollary 2 is our first upper bound on the syntactic complexity of a star-free language with n quotients, but this bound is not tight in general because the set A_n *is not a semigroup* for $n \geqslant 3$. For example, if a: $[1, 3, 1]$ and b: $[2, 2, 1]$, then ab: $[2, 1, 2]$, which contains the cycle $(1, 2)$. Hence our task is to find the size of the largest semigroup contained in A_n.

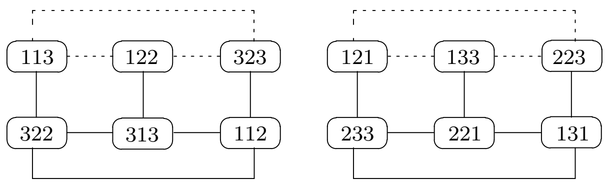

Figure 2: Conflict graph for $n = 3$.

First, let us consider small values of n:

1. If $n = 1$, the only two languages, $\emptyset$ and Σ^*, are star-free, since $\Sigma^* = \overline{\emptyset}$. Here $\sigma(L) = 1$, for both languages, the bound of Corollary 2 holds and is tight.
2. If $n = 2$, $|A_2| = 3$. The only unary languages are ε and $\overline{\varepsilon} = aa^*$, and $\sigma(L) = 1$ for both. For $\Sigma = \{a, b\}$, one verifies that $\sigma(L) \leqslant 2$, and $\Sigma^* a \Sigma^*$ meets this bound. If $\Sigma = \{a, b, c\}$, then $L = \Sigma^* a \overline{\Sigma^* b \Sigma^*}$ has $\sigma(L) = 3$.

In summary, for $n = 1$ and 2, the bound of Corollary 2 is tight for $|\Sigma| \geqslant 1$ and $|\Sigma| \geqslant 3$, respectively.

We say that two aperiodic transformations a and b *conflict* if ab or ba contains a cycle; then (a, b) is called a *conflicting pair*. When $n = 3$, $|A_3| = 4^2 = 16$. The transformations $a_0 = [1, 2, 3]$, $a_1 = [1, 1, 1]$, $a_2 = [2, 2, 2]$, $a_3 = [3, 3, 3]$ cannot create any conflict. Hence we consider only the remaining 12 transformations.

Let $b_1 = [1, 1, 3]$, $b_2 = [1, 2, 1]$, $b_3 = [1, 2, 2]$, $b_4 = [1, 3, 3]$, $b_5 = [2, 2, 3]$, and $b_6 = [3, 2, 3]$. Each of them has only one conflict. There are also two *conflicting triples* (b_1, b_3, b_6) and (b_2, b_4, b_5), since $b_1 b_3 b_6$ and $b_2 b_4 b_5$ both contain a cycle. Figure 2 shows the conflict graph of these 12 transformations, where normal lines indicate conflicting pairs, and dotted lines indicate conflicting triples. To save space we use three digits to represent each transformation, for example, 112 stands for the transformation $[1, 1, 2]$, and $(112)(113) = 111$. We can choose at most two inputs from each triple and at most one from each conflicting pair. So there are at most 6 conflict-free transformations from the 12, for example, b_1, b_3, b_4, b_5, $c_1 = [1, 1, 2]$, $c_2 = [2, 3, 3]$. Adding a_0, a_1, a_2 and a_3, we get a total of at most 10. The inputs a_0, b_4, b_5, c_1 are conflict-free and generate precisely these 10 transformations. Hence $\sigma(L) \leqslant 10$ for any star-free language L with $\kappa(L) = n = 3$, and this bound is tight with $|\Sigma| = 4$.

4. Finite/Cofinite, Reverse Definite, and Definite Languages

One of the simplest classes of regular languages is the class of finite and cofinite languages, where a language is *cofinite* if its complement is finite. We also study two related classes: reverse definite and definite languages.

4.1. Finite/Cofinite Languages

Let L be a regular language and let $\mathcal{D} = (Q, \Sigma, \delta, q_1, F)$ be its minimal DFA. It is well-known that L is finite/cofinite if and only if there exists a numbering $1, \ldots, n$ on

Q such that for all $w \in \Sigma^+$, $\delta(i, w) = j$ implies that $i < j$ or $i = j = n$. Next, we define the set B_n of transformations on Q:

$$B_n = \{t \mid it > i \; \forall \, i = 1, \ldots, n-1, \text{ and } nt = n\}.$$

It is clear that B_n is a semigroup under composition and that its size is $(n-1)!$.

Theorem 3 *Let L be a finite or cofinite language with state complexity n. Then the syntactic complexity of L satisfies $\sigma(L) \leqslant (n-1)!$ and this bound is tight.*

Proof. Let $\mathcal{D} = (Q, \Sigma, \delta, q_1, F)$ be the minimal DFA of L. The above discussion implies that we may label the states Q so that S_L is a subsemigroup of B_n. Therefore the bound holds.

Let $n \geqslant 1$ and $|\Sigma| = (n-1)!$. Let $\mathcal{D}$ be a DFA with states numbered $\{1, 2, \ldots, n\}$, initial state 1, empty state n, and final state $n-1$. For each transformation $t \in B_n$, assign a letter in Σ whose transformation in $\mathcal{D}$ is exactly t. To show that $\mathcal{D}$ is minimal, note that state $i > 1$ is reached from the initial state 1 by the transformation $[i, n, n, \ldots, n]$. Also, if i and j are two states and $i < j \leqslant n-2$, then the transformation $t \in B_n$ that has $it = n-1$, and $kt = n$ for all other $k \neq i$, distinguishes the two states. State $n-1$ is distinguishable from other states because it accepts ε, and state n is the unique empty state. Hence $\mathcal{D}$ is minimal and accepts a finite language. Therefore the bound is tight.

For cofinite languages, interchange final and non-final states and use the same argument. □

A natural question is the minimal size of the alphabet required to achieve the upper bound. Let $\mathcal{D}$ be the minimal DFA of a finite or cofinite language L with $S_L = B_n$. For any state $i \in Q$ and $a \in \Sigma$, it is clear that $\delta(i, a) \geq i + 1$ or $i = n$. It follows that if an input transformation $t \in B_n$ satisfies $it = i + 1$ for *some* $i \in \{1, 2, \ldots, n-2\}$, then any word w corresponding to t must have length 1, that is, w must be in Σ.

Theorem 4 *Let $L \subseteq \Sigma^*$ be a finite or cofinite language with state complexity $n \geqslant 3$, and suppose that $\sigma(L) = (n-1)!$. Then $|\Sigma| \geqslant (n-1)! - (n-2)!$ and this bound is tight.*

Proof. By Theorem 3, we may assume that $S_L = B_n$. The preceding discussion implies that $|\Sigma|$ is at least the number of transformations which satisfy $it = i + 1$ for some $i = 1, \ldots, n-2$. Let $G_n \subset B_n$ be the set of these transformations. If we place the restriction $it \neq i + 1$ for all $i \in \{1, 2, \ldots, n-2\}$, then there are $n - i - 1$ choices for these it, and hence a total of $(n-2)!$ such transformations. Therefore $|G_n| = |B_n| - (n-2)! = (n-1)! - (n-2)!$. Now let $t = [j_1, \ldots, j_{n-2}, n, n] \in B_n$ be arbitrary. We have $j_{n-2} \in \{n-1, n\}$. If $j_{n-2} = n-1$, then $t \in G_n$. Otherwise, $j_{n-2} = n$, and let $t' = [j_1 - 1, \ldots, j_{n-3} - 1, j_{n-2} - 1, n, n] = [j_1 - 1, \ldots, j_{n-3} - 1, n-1, n, n]$. Then $t' \in G_n$ and $t = t'[2, 3, \ldots, n-1, n, n]$. Thus G_n generates B_n, and the bound is tight. □

Example 5 For $n = 4$, the largest transition semigroup of the minimal DFA of a finite language has $(4-1)! = 6$ elements

$$B_4 = \{[\mathbf{2}, \mathbf{3}, \mathbf{4}, \mathbf{4}], [\mathbf{2}, \mathbf{4}, \mathbf{4}, \mathbf{4}], [\mathbf{3}, \mathbf{3}, \mathbf{4}, \mathbf{4}], [3, 4, 4, 4], [\mathbf{4}, \mathbf{3}, \mathbf{4}, \mathbf{4}], [4, 4, 4, 4]\},$$

and its minimal generating set of 4 elements is shown in boldface. ∎

4.2. Reverse Definite Languages

A *reverse definite language* is a language $L \subseteq \Sigma^*$ of the form $L = E \cup F\Sigma^*$, where E and F are finite languages. Because reverse definite languages are characterized by prefixes of a fixed length, their minimal DFAs (and hence syntactic complexity bounds) are very similar to those of finite/cofinite languages. We assume L is not finite in this section. For state complexities $n > 1$, we note first that if $\emptyset$ is not a quotient of L, then L is cofinite. Otherwise, $\emptyset$ and Σ^* are both quotients of L. Let $\mathcal{D} = (Q, \Sigma, \delta, q_1, F)$ be the minimal DFA of L, and label the states corresponding to $\emptyset$ and Σ^* with $n-1$ and n, respectively. One can number the other states in Q so that for all words $w \in \Sigma^+$, if $\delta(i, w) = j$ then $i \leqslant j$, with equality if and only if $i \in \{n-1, n\}$.

The syntactic complexity results for reverse definite languages now follow directly from the results for finite/cofinite languages.

Theorem 6 *Let $L = E \cup F\Sigma^*$ be a reverse definite language with state complexity $n \geqslant 2$. Then $\sigma(L) \leqslant (n-1)!$, and this bound is tight. Moreover, if this language L achieves this upper bound and $n \geqslant 4$, then $|\Sigma| \geqslant (n-1)! - 2(n-2)!$, and this bound is tight.*

Proof. First, if $\emptyset$ is not a quotient of L, then L is cofinite, and hence has the same bounds as in the previous subsection.

Otherwise, let $\mathcal{D}$ be the minimal DFA recognizing L, and let the states be totally ordered as in the preceding discussion. Define the set of transformations analogous to the case of finite/cofinite languages:

$$B'_n = \{t \mid it > i \;\forall\, i = 1, \ldots, n-2,\ (n-1)t = n-1, \text{ and } nt = n\}.$$

Then $S_L \subseteq B'_n$, which a straightfoward calculation shows to be a semigroup. Clearly, $|B'_n| = (n-1)!$, thus proving the bound.

To find a witness, start with the finite witness as in the proofs of Theorems 3 and 4, make all transitions from state $n-1$ to go to $n-1$ itself, and make state n the only final state. The new automaton is minimal.

For the minimal size of the alphabet, we define $G'_n \subset B'_n$ to be the set of transformations t in B'_n satisfying $it = i+1$ for some $i = 1, \ldots, n-3$. As in Section 4.1, these transformations must correspond to individual letters in Σ. The same indirect counting argument shows that for $n \geqslant 4$, $|G'_n| = (n-1)! - 2 \cdot (n-2)!$, hence proving the bound. Now let $t = [j_1, \ldots, j_{n-3}, j_{n-2}, n-1, n] \in B'_n$ be arbitrary. Let $k = \min_{1 \leqslant l \leqslant n-3}\{j_l - l - 1\}$. Let $t' = [j_1 - k, \ldots, j_{n-3} - k, j_{n-2}, n-1, n]$. Then $t' \in G'_n$, and $t = t'[2, 3, \ldots, n-1, n-1, n]^k$. Thus G'_n generates B'_n. Therefore the alphabet-size bound is tight. □

Example 7 For $n = 4$, the finite witness meeting the bound $(n-1)!$ has the transformation set given in Example 5. We modify this set by making $n-1$ the empty state, thus obtaining

$$B_4' = \{[\mathbf{2},\mathbf{3},\mathbf{3},\mathbf{4}], [\mathbf{2},\mathbf{4},\mathbf{3},\mathbf{4}], [3,3,3,4], [3,4,3,4], [4,3,3,4], [4,4,3,4]\},$$

where the generators are in boldface, and state 4 is final. This DFA accepts a reverse definite language, and meets the syntactic complexity and alphabet-size bounds. ■

4.3. Definite Languages

A *definite language* is a language $L \subseteq \Sigma^*$ of the form $L = E \cup \Sigma^* F$, where E and F are finite languages. Like finite/cofinite and reverse definite languages, definite languages are characterized by their transformation semigroups [10]. It follows from the comprehensive theory of definite languages developed by Perles, Rabin and Shamir [24] that every transformation of the minimal DFA of a definite language must be non-permutational, and conversely, if the transformation semigroup of a minimal DFA contains only non-permutational transformations, then the DFA accepts a definite language.

Our goal for this section is to find the maximal size of a non-permutational transformation semigroup, that is, one which contains only non-permutational transformations. There is a straightforward bijection between such transformations on $\{1,\dots,n\}$ and simple labeled forests on $n-1$ nodes, obtained by removing the unique node for which $it = i$. Then Cayley's Theorem [9, 27] shows that there are n^{n-1} non-permutational transformations of $\{1,\dots,n\}$.

Identifying non-permutational transformations is not sufficient to find a syntactic complexity bound, as the set of such transformations does not form a semigroup for $n \geqslant 3$. For example, the composition of $s = [2,3,3]$ and $t = [1,1,2]$ is $st = [1,2,2]$, which is permutational. Two transformations *conflict* if there exists a permutational transformation in the semigroup that they generate.

We exhibit the following sets of non-permutational transformations which do not conflict; they are similar to the semigroup B_n from Section 4.1.

Theorem 8 *Let $n \geqslant 2$, and define the following sets of transformations:*

$$C_{n,k} = \{t \mid it > i \;\forall\; 1 \leqslant i < k, \;\textit{and}\; it = k \;\forall\; i \geqslant k\}, \quad k = 1,2,3,\dots,n.$$

Then the set of transformations $C_n = \bigcup_{k=1}^{n} C_{n,k}$ is a maximal non-permutational semigroup of size $\lfloor \mathbf{e} \cdot (n-1)! \rfloor$.

Proof. One can check that each $C_{n,k}$ is a semigroup. Let $t_i \in C_{n,i}$ and $t_j \in C_{n,j}$, with $i < j$. A direct computation shows that $t_i t_j \in C_{n,it_j}$, and $t_j t_i \in C_{n,i}$; hence C_n is a semigroup. Moreover, for all $t \in C_{n,k}$, $t^{k-1} = \binom{Q}{k}$, and so all of the transformations are non-permutational.

A counting argument shows that $|C_{n,k}| = \frac{(n-1)!}{(n-k)!}$. Since the $C_{n,k}$ are disjoint,

$$|C_n| = \sum_{k=1}^{n} \frac{(n-1)!}{(n-k)!} = \sum_{l=0}^{n-1} \frac{(n-1)!}{(n-1-l)!} = \lfloor \mathbf{e} \cdot (n-1)! \rfloor;$$

this result is well known in combinatorics; see [14], for example.

For the maximality of C_n, we show that adding any other non-permutational transformation creates a conflict. Let $t \notin C_n$ be non-permutational, with $it = i$.

First suppose that there exists a $j < i$ with $jt = k \leqslant j$. Since t is non-permutational, we may assume $k < j$. Then there exists a $t' \in C_{n,i}$ with $kt' = j$; then $itt' = i$ and $jtt' = j$, and so t and t' conflict.

If no such j exists, then there must exist a $j > i$ with $jt \neq i$. Consider the sequence defined by $j_0 = j$, $j_l = j_{l-1}t$. If there exists an l such that $j_l t = j_{l+1} < i$, let l be the minimal one. Let $t' \in C_{n,j_l}$ with $j_{l+1}t' = i$ and $it' = j_l$. Then $itt' = j_l$, $j_l tt' = i$, and so tt' is permutational. Finally suppose $j_l \geqslant i$ for all l. Since t is non-permutational, i must appear in the sequence; moreover, since $j_1 = jt \neq i$, we can pick $l \geqslant 0$ so that $i = j_{l+2}$. Since $j_{l+1} > i$, we may find a transformation $t' \in C_{n,j_l}$ with $it' = j_{l+1}$ and $j_{l+1}t' = j_l$. Then $it't = i$, $j_{l+1}t't = j_{l+1}$, and $t't$ is permutational. □

To compute the generators of C_n, we require the following definition. Let D_n be the set of all transformations $t = [i_1, \ldots, i_n] \in C_n$ with all $i_j < n$. Define the function $\alpha \colon D_n \to C_n$ by $\alpha(t) = [i_1 + 1, \ldots, i_n + 1]$, and also $\alpha(D_n) = \{t \in C_n \mid \alpha(t_0) = t \text{ for some } t_0 \in D_n\}$. Clearly, α is a bijection. Note that we may also write $\alpha(t) = t[2, 3, \ldots, n, n]$.

Theorem 9 *Let $H_n = C_n \backslash \alpha(D_n)$. Then*

(1) H_n is the minimum set of generators for C_n.

(2) $|H_n| = \lfloor \mathbf{e} \cdot (n-1)! \rfloor - \lfloor \mathbf{e} \cdot (n-2)! \rfloor$.

Proof. For (1), note that $[2, 3, \ldots, n, n] \in H_n$. For any $t \in C_n$, we can write $t = t_0[2, 3, \ldots, n, n]^k$ with $k \geqslant 0$ and $t_0 \in H_n$. Therefore H_n generates C_n.

Now let $t_i \in C_{n,i}$ and $t_j \in C_{n,j}$, with $i \geqslant j$. We consider mt_it_j, and use the fact that each transformation $t \in C_{n,k}$ satisfies $mt \geqslant \min\{k, m+1\}$. There are two cases:

(a) If $m \geqslant j - 1$, then $mt_i \geqslant \min\{i, m+1\} \geqslant j$; hence $mt_it_j = j$.

(b) If $m \leqslant j - 2 < i$, then $mt_i \geqslant m + 1$; hence $mt_it_j \geqslant \min\{j, mt_i + 1\} \geq m + 2$.

It follows that $\alpha^{-1}(t_it_j) \in C_{n,j-1}$; a similar argument shows that $\alpha^{-1}(t_jt_i) \in C_{n,jt_i-1}$. Consequently, no transformation in H_n is a composition of two others in C_n, and so H_n is the minimum generating set of C_n.

For (2), we calculate $|\alpha(D_n)|$, or equivalently $|D_n|$ because α is a bijection. A counting argument shows that $|C_{n,k} \cap D_n| = \frac{(n-2)!}{(n-2-(k-1))!}$. Therefore

$$|H_n| = |C_n| - |\alpha(D_n)| = |C_n| - \sum_{k=1}^{n-1} \frac{(n-2)!}{(n-2-(k-1))!} = \lfloor \mathbf{e} \cdot (n-1)! \rfloor - \lfloor \mathbf{e} \cdot (n-2)! \rfloor.$$

□

The following corollary establishes a direct connection with definite languages.

Corollary 10 *For all $n \geqslant 2$, there exists a definite language L with state complexity n, syntactic complexity $\sigma(L) = \lfloor \mathbf{e}\cdot(n-1)! \rfloor$, and alphabet size $\lfloor \mathbf{e}\cdot(n-1)! \rfloor - \lfloor \mathbf{e}\cdot(n-2)! \rfloor$.*

Proof. Let $\mathcal{D} = (Q, \Sigma, \delta, q_1, F)$ be a DFA with $Q = \{1, 2, \ldots, n\}$, $q_1 = 1$, $F = \{n\}$, and $|\Sigma| = \lfloor \mathbf{e} \cdot (n-1)! \rfloor - \lfloor \mathbf{e} \cdot (n-2)! \rfloor$ with each letter representing a different transformation in H_n, so that the transformation semigroup of $\mathcal{D}$ is C_n. We claim that this is a minimal DFA of a definite language. First, all the states are reachable by the constant transformations $(Q \to i) \in C_n$. Also, any two states i, j with $i < j < n$ are distinguishable by the transformation $t \in C_n$ which acts as $kt = k+1$ for $1 \leqslant k \leqslant i$, and $kt = n$ for $k > i$. State n is distinguishable from every other state because it is the only final state. Hence $\mathcal{D}$ is minimal. Since C_n is a non-permutational semigroup, $\mathcal{D}$ accepts a definite language. □

Conjecture 11 *Let L be a definite language with state complexity $n \geqslant 2$. Then $\sigma(L) \leqslant \lfloor \mathbf{e} \cdot (n-1)! \rfloor$, and if equality holds then $|\Sigma| \geqslant \lfloor \mathbf{e} \cdot (n-1)! \rfloor - \lfloor \mathbf{e} \cdot (n-2)! \rfloor$.*

Example 12 For $n = 4$ we have the following transformations in C_n:

$$
\begin{aligned}
C_{4,1} &= \{[\mathbf{1},\mathbf{1},\mathbf{1},\mathbf{1}]\},\\
C_{4,2} &= \{[2,2,2,2], [\mathbf{3},\mathbf{2},\mathbf{2},\mathbf{2}], [\mathbf{4},\mathbf{2},\mathbf{2},\mathbf{2}]\},\\
C_{4,3} &= \{[\mathbf{2},\mathbf{3},\mathbf{3},\mathbf{3}], [\mathbf{2},\mathbf{4},\mathbf{3},\mathbf{3}], [3,3,3,3], [\mathbf{3},\mathbf{4},\mathbf{3},\mathbf{3}], [4,3,3,3], [\mathbf{4},\mathbf{4},\mathbf{3},\mathbf{3}]\},\\
C_{4,4} &= \{[\mathbf{2},\mathbf{3},\mathbf{4},\mathbf{4}], [\mathbf{2},\mathbf{4},\mathbf{4},\mathbf{4}], [\mathbf{3},\mathbf{3},\mathbf{4},\mathbf{4}], [3,4,4,4], [\mathbf{4},\mathbf{3},\mathbf{4},\mathbf{4}], [4,4,4,4]\}.
\end{aligned}
$$

The generators are in boldface. ■

5. Monotonicity in Transformations, Automata and Languages

We now study syntactic semigroups of languages of monotonic and related automata.

5.1. Monotonic Transformations, DFAs and Languages

In Section 3 we have shown that, when $n = 3$, the tight upper bound on the syntactic complexity of star-free languages is 10, and it turns out that this bound is met by a monotonic language (defined below). This provides one reason to study monotonic automata and languages. A second reason is the fact that all the tight upper bounds on the quotient/state complexity of operations on star-free languages are met by monotonic languages [7].

Let $\leqslant$ be a total order on Q. A transformation t of Q is *monotonic* if, for all $p, q \in Q$, $p \leqslant q$ implies $pt \leqslant qt$. From now on we assume that $\leqslant$ is the usual order on integers, and that $p < q$ means that $p \leqslant q$ and $p \neq q$.

Let M_Q be the set of all monotonic transformations of Q. In the following, we restate slightly the result of Gomes and Howie [13, 16] for our purposes, since the work in [13] does not consider the identity transformation to be monotonic.

Theorem 13 (Gomes and Howie) *When $n \geqslant 1$, the set M_Q is an aperiodic semigroup of cardinality*

$$|M_Q| = f(n) = \sum_{k=1}^{n} \binom{n-1}{k-1}\binom{n}{k} = \binom{2n-1}{n},$$

and it is generated by the set $H = \{a, b_1, \ldots, b_{n-1}, c\}$, where, for $1 \leqslant i \leqslant n-1$,

1. *$1a = 1$, $ja = j-1$ for $2 \leqslant j \leqslant n$;*
2. *$ib_i = i+1$, and $jb_i = j$ for all $j \neq i$;*
3. *c is the identity transformation.*

Moreover, for $n = 1$, a and c coincide, but the cardinality of the generating set cannot be reduced for $n \geqslant 2$.

Example 14 For $n = 1$ there is only one transformation $a = c = [1]$ and it is monotonic. For $n = 2$, the three generators are $a = [1,1]$, $b_1 = [2,2]$ and $c = [1,2]$, and M_Q consists of these three transformations. For $n = 3$, the four generators $a = [1,1,2]$, $b_1 = [2,2,3]$, $b_2 = [1,3,3]$, and $c = [1,2,3]$ generate all ten monotonic transformations, as shown in Table 1. For $n = 4$, the following five transformations generate all 35 monotonic transformations: $[1,1,2,3]$, $[2,2,3,4]$, $[1,3,3,4]$, $[1,2,4,4]$ and $[1,2,3,4]$. ■

Table 1: The monotonic transformations for $n = 3$.

	a	b_1	b_2	c	aa	ab_1	ab_2	b_2a	b_2b_1	ab_1b_2
1	1	2	1	1	1	2	1	1	2	3
2	1	2	3	2	1	2	1	2	3	3
3	2	3	3	3	1	2	3	2	3	3

Remark 15 By Stirling's approximation, $f(n) = |M_Q|$ grows asymptotically like $4^n/(2\sqrt{\pi n})$ as $n \to \infty$.

Now we turn to DFAs whose inputs perform monotonic transformations. A DFA is *monotonic* [1] if all transformations in its transition semigroup are monotonic with respect to some fixed total order. Every monotonic DFA is aperiodic because monotonic transformations are aperiodic. A regular language is *monotonic* if its minimal DFA is monotonic.

Let us now define a DFA having as inputs the generators of M_Q:

Definition 16 *For $n \geqslant 1$, let $\mathcal{D}_n^{\mathrm{M}} = (Q, \Sigma, \delta, 1, \{n\})$ be the DFA in which $Q = \{1, \ldots, n\}$, $\Sigma = \{a, b_1, \ldots, b_{n-1}, c\}$, and each letter in Σ performs the transformation defined in Theorem 13.*

DFA $\mathcal{D}_n^{\rm M}$ is minimal for the following reasons: For $2 \leqslant i \leqslant n$, state i is reached by $w_i = b_1 \cdots b_{i-1}$; so all states are reachable. For all i, j such that $1 \leqslant i < j < n$, state j accepts $u_j = b_j \cdots b_{n-1}$, but state i rejects u_j; state n is the unique final state. Hence all states are distinguishable. From Theorem 13 we have

Corollary 17 *For $n \geqslant 1$, the syntactic complexity $\sigma(L)$ of any monotonic language L with state complexity n satisfies $\sigma(L) \leqslant f(n) = \binom{2n-1}{n}$. Moreover, this bound is met by the language $L(\mathcal{D}_n^{\rm M})$ of Definition 16, and when $n \geqslant 2$ it cannot be met by any monotonic language over an alphabet having fewer than $n+1$ letters.*

5.2. Monotonic Partial Transformations and IDFAs

As we shall see, for $n \geqslant 4$ the maximal syntactic complexity cannot be reached by monotonic languages. We now extend the concept of monotonicity from full transformations to partial transformations, and hence define a new subclass of star-free languages. The upper bound of syntactic complexity of languages in this subclass is above that of monotonic languages for $n \geqslant 4$.

Let $\leqslant$ be a total order on Q. A partial transformation t of Q is *monotonic* if, for all $p, q \in \mathrm{dom}(t)$, $p \leqslant q$ implies $pt \leqslant qt$. As before, we assume that the total order on Q is the usual order on integers. Let PM_Q be the set of all monotonic partial transformations of Q with respect to such an order. Gomes and Howie [13] showed the following result, again restated slightly:

Theorem 18 (Gomes and Howie) *When $n \geqslant 1$, the set PM_Q is an aperiodic semigroup of cardinality*

$$|PM_Q| = g(n) = \sum_{k=0}^{n} \binom{n}{k} \binom{n+k-1}{k},$$

and it is generated by $I = \{a, b_1, \ldots, b_{n-1}, c_1, \ldots, c_{n-1}, d\}$, where, for $1 \leqslant i \leqslant n-1$,

1. *$1a = \Box$, and $ja = j-1$ for $j = 2, \ldots, n$;*
2. *$ib_i = i+1$, $(i+1)b_i = \Box$, and $jb_i = j$ for $j = 1, \ldots, i-1, i+2, \ldots, n$;*
3. *$ic_i = i+1$, and $jc_i = j$ for all $j \neq i$;*
4. *d is the identity transformation.*

Moreover, the cardinality of the generating set cannot be reduced.

Example 19 For $n = 1$, the two monotonic partial transformations are $a = [\Box]$, and $d = [1]$. For $n = 2$, the eight monotonic partial transformations are generated by $a = [\Box, 1]$, $b_1 = [2, \Box]$, $c_1 = [2, 2]$, and $d = [1, 2]$. For $n = 3$, the 38 monotonic partial transformations are generated by $a = [\Box, 1, 2]$, $b_1 = [2, \Box, 3]$, $b_2 = [1, 3, \Box]$, $c_1 = [2, 2, 3]$, $c_2 = [1, 3, 3]$ and $d = [1, 2, 3]$.

Partial transformations correspond to IDFAs. For example, $a = [\Box, 1]$, $b = [2, \Box]$ and $c = [2, 2]$ correspond to the transitions of the IDFA of Figure 3 (a). ■

Laradji and Umar [19] proved the following asymptotic approximation:

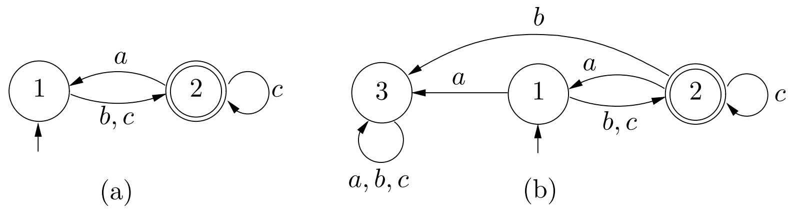

Figure 3: Partially monotonic automata: (a) IDFA; (b) DFA $\mathcal{D}_0$.

Remark 20 For large n, $g(n) = |PM_Q| \sim 2^{-3/4}(\sqrt{2}+1)^{2n+1}/\sqrt{\pi n}$.

An IDFA is *monotonic* if all partial transformations in its transition semigroup are monotonic with respect to some fixed total order. A minimal DFA is *partially monotonic* if its corresponding minimal IDFA is monotonic. A regular language is *partially monotonic* if its minimal DFA is partially monotonic. Note that monotonic languages are also partially monotonic.

Example 21 If we complete the transformations in Figure 3 (a) by replacing the undefined entry $\square$ by a new empty state 3, as usual, we obtain the DFA of Figure 3 (b). That DFA is not monotonic, because $1 < 2$ implies $2 < 3$ under input b and $3 < 2$ under ab. A contradiction is also obtained if we assume that $2 < 1$. However, this DFA is partially monotonic, since its corresponding IDFA, shown in Figure 3 (a), is monotonic.

The DFA of Figure 4 is monotonic for the order shown. It has an empty state, and is also partially monotonic for the same order. ■

Consider any partially monotonic language L with quotient complexity n. If its minimal DFA $\mathcal{D}$ does not have the empty quotient, then L is monotonic; otherwise, its minimal IDFA $\mathcal{I}$ has $n-1$ states, and the transition semigroup of $\mathcal{I}$ is a subset of $PM_{Q'}$, where $Q' = \{1, \dots, n-1\}$. Hence we consider the following semigroup CM_Q of *monotonic completed transformations* of Q. Start with the semigroup $PM_{Q'}$. Convert all $t \in PM_{Q'}$ to full transformations by adding n to $\text{dom}(t)$ and letting $it = n$ for all $i \in Q \setminus \text{dom}(t)$. Such a conversion provides a one-to-one correspondence between $PM_{Q'}$ and CM_Q. For $n \geqslant 2$, let $e(n) = g(n-1)$. Then semigroups CM_Q and $PM_{Q'}$

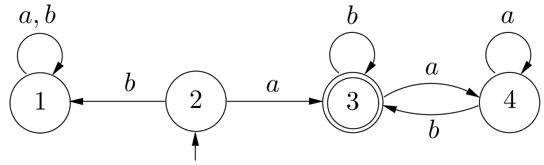

Figure 4: Partially monotonic DFA that is monotonic and has an empty state.

are isomorphic, and $e(n) = |CM_Q|$.

Definition 22 *For $n \geqslant 2$, let $\mathcal{D}_n^{\mathrm{PM}} = (Q, \Sigma, \delta, 1, \{n-1\})$ be the DFA in which $Q = \{1, \dots, n\}$, $\Sigma = \{a, b_1, \dots, b_{n-2}, c_1, \dots, c_{n-2}, d\}$, and each letter in Σ defines a transformation such that, for $1 \leqslant i \leqslant n-2$,*

1. *$1a = na = n$, and $ja = j-1$ for $j = 2, \dots, n-1$;*
2. *$ib_i = i+1$, $(i+1)b_i = n$, and $jb_i = j$ for $j = 1, \dots, i-1, i+2, \dots, n$;*
3. *$ic_i = i+1$, and $jc_i = j$ for all $j \neq i$;*
4. *d is the identity transformation.*

DFA $\mathcal{D}_n^{\mathrm{PM}}$ is minimal for the following reasons: For $2 \leqslant i \leqslant n-1$, state i is reached by $w_i = b_1 \cdots b_{i-1}$, and state n is reached by a; hence all states are reachable. For $1 \leqslant i \leqslant n-2$ only state i accepts $b_i \cdots b_{n-2}$, state $n-1$ is the unique final state, and state n is the unique empty state. Hence all the states are distinguishable.

We know that monotonic languages are also partially monotonic. As shown in Table 2 on page 21, $|M_Q| = f(n) > e(n) = |CM_Q|$ for $n \in \{2, 3\}$. On the other hand, one verifies that $e(n) > f(n)$ when $n \geqslant 4$. By Corollary 17 and Theorem 18, we have

Corollary 23 *The syntactic complexity of a partially monotonic language L with state complexity n satisfies $\sigma(L) \leqslant f(n)$ for $n \leqslant 3$, and $\sigma(L) \leqslant e(n)$ for $n \geqslant 4$. Moreover, when $n \geqslant 4$, this bound is met by $L(\mathcal{D}_n^{\mathrm{PM}})$ of Definition 22, and it cannot be met by any partially monotonic language over an alphabet having fewer than $2n-2$ letters.*

Table 2 contains these upper bounds for small values of n. By Remark 20, the upper bound $e(n)$ is asymptotically $2^{-3/4}(\sqrt{2}+1)^{2n-1}/\sqrt{\pi(n-1)}$.

5.3. Nearly Monotonic Transformations and DFAs

In this section we develop an even larger aperiodic semigroup based on partially monotonic languages.

Let K_Q be the set of all constant transformations of Q, and let $NM_Q = CM_Q \cup K_Q$. We shall call the transformations in NM_Q *nearly monotonic* with respect to the usual order on integers.

Theorem 24 *When $n \geqslant 2$, the set NM_Q of all nearly monotonic transformations of a set Q of n elements is an aperiodic semigroup of cardinality*

$$|NM_Q| = h(n) = e(n) + (n-1) = \sum_{k=0}^{n-1} \binom{n-1}{k} \binom{n+k-2}{k} + (n-1),$$

and it is generated by the set $J = \{a, b_1, \dots, b_{n-2}, c_1, \dots, c_{n-2}, d, e\}$ of $2n-1$ transformations of Q, where e is the constant transformation $(Q \to 1)$, and all other transformations are as in Definition 22. Moreover, the cardinality of the generating set cannot be reduced.

Proof. Pick any $t_1, t_2 \in NM_Q$. If $t_1, t_2 \in CM_Q$, then $t_1t_2 \in CM_Q$. Otherwise $t_1 \in K_Q$ or $t_2 \in K_Q$, and t_1t_2 is a constant transformation. Hence $t_1t_2 \in NM_Q$ and NM_Q is a semigroup. Since constant transformations are aperiodic and CM_Q is aperiodic, NM_Q is also aperiodic.

If X is a set of transformations, let $\langle X\rangle$ denote the semigroup generated by X. Since $J \subseteq NM_Q$, $\langle J\rangle \subseteq NM_Q$. Let $I' = J \setminus \{e\}$, and $Q' = Q \setminus \{n\}$. Then $PM_{Q'} \simeq CM_Q = \langle I'\rangle$. For any $t = (Q \to j) \in K_Q$, where $j \in Q$, since $s_j = [j, \ldots, j, n] \in CM_Q \subseteq \langle J\rangle$, we have $t = es_j \in \langle J\rangle$. So $NM_Q = \langle J\rangle$. Note that $(Q \to i) \in CM_Q$ if and only if $i = n$. Thus $h(n) = |NM_Q| = |PM_{Q'}| + (n-1) = e(n) + (n-1)$.

Suppose J' is a generating set of NM_Q and $|J'| < |J|$. Note that $NM_Q = CM_Q \cup K_Q$, and $CM_Q \cap K_Q = \{(Q \to n)\}$. Let $R = J' \cap CM_Q$. We must have $J' \setminus R \neq \emptyset$; otherwise, since CM_Q is a semigroup, $J' = R \subseteq CM_Q$, and $\langle J'\rangle \subseteq CM_Q \neq NM_Q$. So $|R| < |I'|$. Pick any $t \in CM_Q$. Since $CM_Q \subset NM_Q = \langle J'\rangle$, there exist $s_1, \ldots, s_k \in J'$ such that $t = s_1 \cdots s_k$. If $t \neq (Q \to n)$, then $t \notin K_Q$; since K_Q is an ideal of NM_Q, we have $s_i \in J' \setminus K_Q \subseteq R$ for all s_i, and $t \in \langle R\rangle$. If $t = (Q \to n)$, then $t = [2, 3, \ldots, n, n]^{n-1} \in \langle R\rangle$ as well. Thus $\langle R\rangle = CM_Q$. This contradicts the fact that I' is a minimal generating set of CM_Q. So J is a minimal generating set of NM_Q. □

Example 25 For $n = 2$, the three nearly monotonic transformations are $a = [2, 2]$, $d = [1, 2]$ and $e = [1, 1]$. For $n = 3$, the ten nearly monotonic transformations are generated by $a = [3, 1, 3]$, $b_1 = [2, 3, 3]$, $c_1 = [2, 2, 3]$, $d = [1, 2, 3]$, and $e = [1, 1, 1]$. ■

An input $a \in \Sigma$ is *constant* if it performs a constant transformation of Q. Let $\mathcal{D}$ be a DFA with alphabet Σ; then $\mathcal{D}$ is *nearly monotonic* if, after removing constant inputs, the resulting DFA $\mathcal{D}'$ is partially monotonic. A regular language is *nearly monotonic* if its minimal DFA is nearly monotonic.

Definition 26 *For $n \geqslant 2$, let $\mathcal{D}_n^{\mathrm{NM}} = (Q, \Sigma, \delta, 1, \{n-1\})$ be a DFA, where $Q = \{1, \ldots, n\}$, $\Sigma = \{a, b_1, \ldots, b_{n-2}, c_1, \ldots, c_{n-2}, d, e\}$, and each letter in Σ performs the transformation defined in Definition 22 and Theorem 24.*

Theorem 24 now leads to the following result:

Theorem 27 *For $n \geqslant 2$, if L is a nearly monotonic language with state complexity n, then $\sigma(L) \leqslant h(n) = \sum_{k=0}^{n-1} \binom{n-1}{k}\binom{n+k-2}{k} + (n-1)$. Moreover, this bound is met by the language $L(\mathcal{D}_n^{\mathrm{NM}})$ of Definition 26, and cannot be met by any nearly monotonic language over an alphabet having fewer than $2n-1$ letters.*

Proof. Note that $\mathcal{D}_n^{\mathrm{NM}}$ is obtained from $\mathcal{D}_n^{\mathrm{PM}}$ by adding the input e. Since $\mathcal{D}_n^{\mathrm{PM}}$ is minimal, $\mathcal{D}_n^{\mathrm{NM}}$ is minimal as well. Thus L has state complexity n. The syntactic semigroup of L is generated by J; so L has syntactic complexity $\sigma(L) = h(n) = \sum_{k=0}^{n-1} \binom{n-1}{k}\binom{n+k-2}{k} + (n-1)$, and it is nearly monotonic. □

Recall that $e(n) > f(n)$ for $n \geqslant 4$. Since $h(n) = e(n) + (n-1)$, and $h(n) = f(n)$ for $n \in \{2, 3\}$, as shown in Table 2, we have $h(n) \geqslant f(n)$ for $n \geqslant 2$, and the maximal syntactic complexity of nearly monotonic languages is at least that of both monotonic and partially monotonic languages.

Although we cannot prove that NM_Q is the largest semigroup of aperiodic transformations, we can show that no transformation can be added to NM_Q. The following result shows that the syntactic semigroup $S_{L(\mathcal{D}_n^{\mathrm{NM}})} = NM_Q$ of $L(\mathcal{D}_n^{\mathrm{NM}})$ in Definition 26 is a local maximum among aperiodic subsemigroups of $\mathcal{T}_Q$.

Proposition 28 *Let S be an aperiodic subsemigroup of $\mathcal{T}_Q$. If $NM_Q \subseteq S$, then $NM_Q = S$.*

Proof. Suppose $S \subseteq \mathcal{T}_Q$ is an aperiodic semigroup and $NM_Q \subseteq S$. Assume $S \neq NM_Q$. Then there exists $t \in S \setminus NM_Q$. There are two cases:

1. $nt = n$. Since $CM_Q \subseteq NM_Q$, t is not a monotonic completed transformation, and there exist $i, j \in Q \setminus \{n\}$ such that $i < j$ and $n > it > jt$. Let $\tau \in \mathcal{T}_Q$ be such that $(jt)\tau = i$, $(it)\tau = j$, and $h\tau = n$ for all $h \neq it, jt$. Then $\tau \in NM_Q \subseteq S$, and $t\tau \in S$. However, $t\tau$ contains a cycle (i, j); so it is not aperiodic.
2. $nt \neq n$. Since $K_Q \subseteq NM_Q$, t is not constant; so there exists $i \neq n$ such that $it \neq nt$. Let $\tau' \in \mathcal{T}_Q$ be such that $(it)\tau' = n$, and $h\tau' = i$ for all $h \neq it$. Then $\tau' \in CM_Q \subseteq NM_Q \subseteq S$, and $t\tau' \in S$. However, $t\tau'$ contains a cycle (i, n); so it is not aperiodic.

Since S is not aperiodic in both cases, we have a contradiction. Hence $NM_Q = S$. □

5.4. Containment and Closure Properties

Let $\mathbb{L}_M$, $\mathbb{L}_{PM}$, $\mathbb{L}_{NM}$, and $\mathbb{L}_{SF}$ be the classes of monotonic, partially monotonic, nearly monotonic and star-free languages. Then the following holds:

Proposition 29 $\mathbb{L}_M \subsetneq \mathbb{L}_{PM} \subsetneq \mathbb{L}_{NM} \subsetneq \mathbb{L}_{SF}$.

Proof. The DFA $\mathcal{D}_0$ of Figure 3 (b) is partially monotonic but not monotonic. If we add to that DFA the input d: $[1, 1, 1]$, then the new DFA is nearly monotonic but not partially monotonic. The DFA of Figure 5 is aperiodic, as one can verify. It has no constant input; hence it must be partially monotonic if it is nearly monotonic. Since it has no empty state, it must be monotonic if it is partially monotonic. However, if $1 < 2$, then $3 < 2$ by a, and also $2 < 3$ by b. We get a similar contradiction if we set $2 < 1$. Therefore $\mathcal{D}$ is not monotonic. One verifies that these DFAs are minimal. □

Proposition 30 *The following closure properties of $\mathbb{L}_M$, $\mathbb{L}_{PM}$, and $\mathbb{L}_{NM}$ hold:*

1. *All three classes are closed under left quotients.*
2. *The class $\mathbb{L}_M$ is closed under complementation, but $\mathbb{L}_{PM}$ and $\mathbb{L}_{NM}$ are not.*
3. *None of the three classes is closed under union, symmetric difference, intersection, difference, product, star and reversal.*

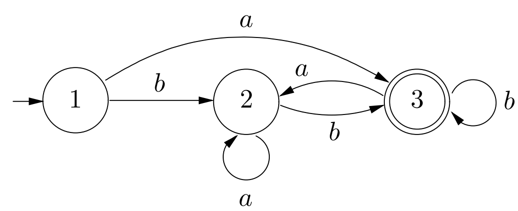

Figure 5: A DFA that is aperiodic but not monotonic.

Proof. The left quotient of any monotonic, partially monotonic, or nearly monotonic language is defined by a DFA that is a subautomaton of the DFA of the original language; such a DFA is minimal, and closure under quotients follows.

For monotonic languages, closure under complementation is obvious, since monotonicity does not involve final states. For partially monotonic languages, consider the DFA $\mathcal{D}_0$ of Figure 3 (b), which is minimal, partially monotonic, and also nearly monotonic since it has no constant inputs. By interchanging final and non-final states, we obtain DFA $\mathcal{D}'_0$, which is minimal and accepts $\overline{L(\mathcal{D}_0)}$. From the proof of Proposition 29 we know that $\mathcal{D}_0$ is not monotonic, and so neither is $\mathcal{D}'_0$; since $\mathcal{D}'_0$ has no empty state, it is not partially monotonic. Hence $\mathbb{L}_{PM}$ is not closed under complement. Because $\mathcal{D}'_0$ does not have any constant inputs, it is not nearly monotonic. Since $\mathcal{D}_0$ is nearly monotonic, $\mathbb{L}_{NM}$ is also not closed under complement.

The language of the DFA of Figure 5 is not nearly monotonic. However, it is the union of languages $L_1 = a(\varepsilon \cup \Sigma^* b)$ and $L_2 = b\Sigma^* b$. The DFA of L_1 is shown in Figure 4, and it is monotonic. One verifies that L_2 is also monotonic. This proves that each of the classes is not closed under union.

Since L_1 and L_2 are disjoint, none of the three classes is closed under symmetric difference, since there is lack of closure under union.

For intersection, consider $L'_1 = \overline{L_1}$ and $L'_2 = \overline{L_2}$; L'_1 and L'_2 are monotonic, since monotonic languages are closed under complement. Since neither L'_1 nor L'_2 has the empty quotient, both are also partially monotonic. Since their DFAs do not have empty quotients, L'_1 nor L'_2 are also nearly monotonic. If $L' = L'_1 \cap L'_2$, then $L' = \overline{L_1} \cap \overline{L_2} = \overline{(L_1 \cup L_2)}$. Since $L_1 \cup L_2$ is not monotonic, neither is L'. Since L' does not have the empty quotient, it is not partially monotonic. Also, the DFA of L' has no constant inputs, and so L' is not nearly monotonic. This proves lack of closure under intersection for all three classes.

Since the class of monotonic languages is closed under complement, it cannot be closed under difference. If the class of partially or nearly monotonic languages were closed under difference, then it would also be closed under complement, since $\overline{L} = \Sigma^* \setminus L$, and Σ^* is in the class.

Now consider DFAs: $\mathcal{D}_3 = (\{1,2,3\}, \{a,b\}, \delta_3, 1, \{2\})$, where $a\colon [1,1,1]$ and $b\colon [2,3,3]$, and $\mathcal{D}_4 = (\{1,2,3\}, \{a,b\}, \delta_4, 1, \{2\})$, where $a\colon [1,1,3]$ and $b\colon [2,3,3]$. Both $\mathcal{D}_3$ and $\mathcal{D}_4$ are monotonic. Let $L = L(\mathcal{D}_3) \cdot L(\mathcal{D}_4)$. Then L is star-free with minimal DFA $\mathcal{D} = (\{1,2,3,4,5,6\}, \{a,b\}, \delta, 1, \{3,6\})$, where $a\colon [1,5,5,1,5,5]$

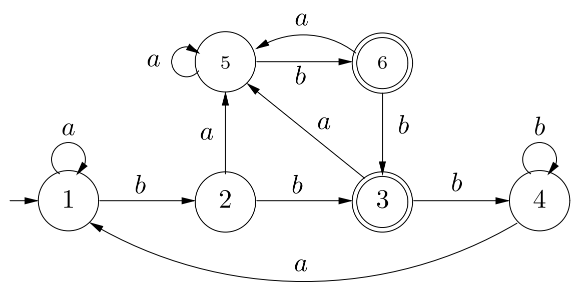

Figure 6: Minimal DFA $\mathcal{D}$ of product of partially monotonic DFAs $\mathcal{D}_3$ and $\mathcal{D}_4$.

and b: $[2, 3, 4, 4, 6, 3]$, as shown in Figure 6. However, $\mathcal{D}$ is not nearly monotonic. If $1 < 6$, then $1 < 5$ by a, $2 < 3$ by b, $3 < 4$ by b again, and $5 < 1$ by a, which is a contradiction. We get a similar contradiction by assuming $6 < 1$. Hence none of the three classes of monotonic languages is closed under product.

Lack of closure under star follows, since each class contains the language $\{aa\}$, and the star of $\{aa\}$ is not star-free.

The counterexamples to closure under reversal are $L(\mathcal{D}_3)$ above for monotonic languages, and $L(\mathcal{D}_5)$ for both partially and nearly monotonic languages, where $\mathcal{D}_5 = (\{1, 2, 3, 4\}, \{a, b\}, \delta_5, 1, \{4\})$, a: $[2, 3, 4, 4]$ and b: $[1, 4, 2, 4]$. □

6. Conclusions

We have found tight upper bounds on the syntactic complexity of finite/cofinite and reverse definite languages. We have conjectured the bounds on the syntactic complexity and the corresponding alphabet size for definite languages. The conjecture has been verified through enumeration for $n \leqslant 4$, but remains unproven for $n > 4$.

Our results on the other three subclasses of star-free languages are summarized in Table 2, where $Q = \{1, \ldots, n\}$, and $Q' = Q \setminus \{n\}$. The numbers in bold type are tight bounds verified using *GAP* [11], by enumerating aperiodic subsemigroups of $\mathcal{T}_Q$. The asterisk $*$ indicates that the bound is already tight for a smaller alphabet. The last four rows show the values of $f(n) = |M_Q|$, $e(n) = |CM_Q| = |PM_{Q'}|$, $h(n) = |NM_Q|$, and the weak upper bound $(n+1)^{n-1}$.

For future work, it would be interesting to establish tight upper bounds on the syntactic complexity of these subclasses for fixed alphabet sizes. But the main open problem is that of syntactic complexity of arbitrary star-free languages.

Acknowledgment

We thank Zoltan Ésik and Judit Nagy-Gyorgy for pointing out to us the relationship between aperiodic inputs and Cayley's theorem. We are grateful to the referees for their many constructive comments.

Table 2: Syntactic complexity of monotonic and related languages.

$\lvert\Sigma\rvert$ n	1	2	3	4	5	6
1	**1**	**1**	**2**	**3**	**5**	**6**
2	$*$	**2**	**7**	**19**	**62**	?
3	$*$	**3**	**9**	**31**	?	?
4	$*$	$*$	**10**	**34**	?	?
5	$*$	$*$	$*$	37	125	?
...	...	...	...	...	...	...
$f(n) = \lvert M_Q\rvert$	**1**	**3**	**10**	35	126	462
$e(n) = \lvert CM_Q\rvert = g(n-1) = \lvert PM_{Q'}\rvert$	–	2	8	38	192	$1,002$
$h(n) = \lvert NM_Q\rvert = e(n) + (n-1)$	–	**3**	**10**	41	196	$1,007$
$(n+1)^{n-1}$	1	3	16	125	$1,296$	$16,807$

References

[1] D. Ananichev, M. Volkov, Synchronizing generalized monotonic automata. *Theoret. Comput. Sci.* **330** (2005) 1, 3–13.

[2] H. Bordihn, M. Holzer, M. Kutrib, Determination of finite automata accepting subregular languages. *Theoret. Comput. Sci.* **410** (2009), 3209–3249.

[3] J. Brzozowski, Canonical regular expressions and minimal state graphs for definite events. In: *Mathematical theory of automata*. Volume 12 of MRI Symposia Series, Polytechnic Institute of Brooklyn, N.Y., 1963, 529–561.

[4] J. Brzozowski, Quotient complexity of regular languages. *J. Autom. Lang. Comb.* **15** (2010) 1/2, 71–89.

[5] J. Brzozowski, B. Li, Syntactic complexity of $\mathcal{R}$- and $\mathcal{J}$-trivial languages. In: H. Jürgensen, R. Reis (eds.), *DCFS 2013*. LNCS 8031, Springer, 2013, 160–171.

[6] J. Brzozowski, B. Li, Y. Ye, Syntactic complexity of prefix-, suffix-, bifix-, and factor-free regular languages. *Theoret. Comput. Sci.* **449** (2012), 37 – 53.

[7] J. Brzozowski, B. Liu, Quotient complexity of star-free languages. In: P. Dömösi, I. Szabolcs (eds.), *13th International Conference on Automata and Formal Languages (AFL)*. Institute of Mathematics and Informatics, College of Nyíregyháza, Debrecen, Hungary, 2011, 138–152. Also to appear in *Int. J. Found. Comput. Sci.*

[8] J. Brzozowski, Y. Ye, Syntactic complexity of ideal and closed languages. In: G. Mauri, A. Leporati (eds.), *DLT 2011*. LNCS 6795, Springer Berlin / Heidelberg, 2011, 117–128.

[9] A. Cayley, A theorem on trees. *Quart. J. Math.* **23** (1889), 376–378.

[10] S. Eilenberg, *Automata, Languages, and Machines*. B, Academic Press, 1976.

[11] GAP-Group, *GAP - Groups, Algorithms, Programming - a System for Computational Discrete Algebra*. 2010. http://www.gap-system.org/.

[12] A. Ginzburg, Abour some properties of definite, reverse definite and related automata. *IEEE Trans. Electronic Comput.* **EC–15** (1966), 806–810.

[13] G. Gomes, J. Howie, On the ranks of certain semigroups of order-preserving transformations. *Semigroup Forum* **45** (1992), 272–282.

[14] L. Halbeisen, N. Hungerbühler, Number theoretic aspects of a combinatorial function. *Notes on Number Theory and Discrete Mathematics* **5** (1999) 4, 138 – 150.

[15] M. Holzer, B. König, On deterministic finite automata and syntactic monoid size. *Theoret. Comput. Sci.* **327** (2004) 3, 319–347.

[16] J. M. Howie, Products of idempotents in certain semigroups of transformations. *Proc. Edinburgh Math. Soc.* **17** (1971) 2, 223–236.

[17] S. C. Kleene, Representation of events in nerve nets and finite automata. In: C. E. Shannon, J. McCarthy (eds.), *Automata Studies*. Princeton University Press, Princeton, NJ, 1954, 3–41.

[18] B. Krawetz, J. Lawrence, J. Shallit, State complexity and the monoid of transformations of a finite set. 2003. http://arxiv.org/abs/math/0306416v1.

[19] A. Laradji, A. Umar, Asymptotic results for semigroups of order-preserving partial transformations. *Comm. Alg.* **34** (2006) 3, 1071–1075.

[20] A. N. Maslov, Estimates of the number of states of finite automata. *Dokl. Akad. Nauk SSSR* **194** (1970), 1266–1268 (Russian). English translation: Soviet Math. Dokl. 11 (1970), 1373–1375.

[21] R. McNaughton, S. A. Papert, *Counter-Free Automata*. M.I.T. Research Monographs 65, The MIT Press, 1971.

[22] J. Myhill, Finite automata and the representation of events. *Wright Air Development Center Technical Report* **57–624** (1957).

[23] A. Nerode, Linear automaton transformations. *Proc. Amer. Math. Soc.* **9** (1958), 541–544.

[24] M. Perles, M. O. Rabin, E. Shamir, The theory of definite automata. *IEEE Trans. Electronic Comput.* **EC–12** (1963), 233–243.

[25] G. Rozenberg, A. Salomaa (eds.), *Handbook of Formal Languages, vol. 1: Word, Language, Grammar*. Springer, New York, NY, USA, 1997.

[26] M. Schützenberger, On finite monoids having only trivial subgroups. *Inform. and Control* **8** (1965), 190–194.

[27] P. W. Shor, A new proof of Cayley's formula for counting labeled trees. *J. Combin. Theory Ser. A* **71** (1995) 1, 154–158.

[28] S. Yu, State complexity of regular languages. *J. Autom. Lang. Comb.* **6** (2001), 221–234.